Nr 9-10/1993 INDEKS Nr 32034 X ISSN 1230-7947

Commodore

NUMER ŚWIĄTECZNY

AMIGOWIEC

Pismo użytkowników komputerów AMIGA

cena 25.000 zł

TARGI W KOLONII
BRILLIANCE
VISTA PRO 3.5

WARSZTATY:
DTP
DTV
GRAFIKA
DOS
I INNE

AMIGA

80 STRON





Targi w Kolonii

Tomasz Kulbacki, Wojciech Białkowski, Ryszard Kowalski

Od 5 do 7 listopada na terenach wystawowych w Köln odbyły się kolejne targi komputerowe World of Commodore. Na tak prestiżowym show nie mogło oczywiście zabraknąć reprezentacji Amigowca w najsilniejszym składzie.

W przeciwieństwie do poprzednich imprez tego rodzaju, Firma ICP, której Commodore powierzył organizację imprezy, zdecydowała się na połączenie w całość trzech odrębnych przedsięwzięć: World of Commodore, World of Games i Computer '93. W zamyśle organizatorów, targi Computer '93 miały połączyć oferty producentów sprzętu do wszystkich systemów komputerowych. Efektem tych ambitnych zamierzeń było jedno spore stoisko, na którym, po dość nieprzyjemnych cenach, sprzedawano wszystko, co da się upakować w obudowie PC. Podobnie trudno zrozumieć, dlaczego zespół kilku stoisk sprzedających cartridge z grami do różnych konsoli nazwano dumnie World of Games. Pozostała część hali o powierzchni ok. 11000 metrów kwadratowych została zajęta przez ponad 150 firm oferujących osprzęt i oprogramowanie do wszystkich mutacji Amigi. Pośrodku królowało oczywiście stoisko Commodore o powierzchni 350 metrów kwadratowych.

Targi rozpoczęły się w piątek, o godzinie dziewiątej, inauguracyjnym przemówieniem zaproszonego przez organizatorów przedstawiciela rządu RFN. Na konferencji prasowej, która odbyła się wkrótce potem, aktualny szef niemieckiego oddziału Commodore p. Alwin Stumpf przedstawił aktualną sytuację firmy, jej politykę i plany na przyszłość. Jak wiadomo, w ciągu ostatnich kilku miesięcy w polityce firmy zaszły istotne zmiany. Commodore w niedalekiej przeszłości był liczącym się producentem komputerów PC-kompatybilnych. W roku 1986 (czyli w roku wejścia na rynek Amigi 1000) firma była największym (ponad 300 tys. sztuk rocznie) wytwórcą tego sprzętu w Europie. Później zaczęły się kłopoty. Konkurencja tanich wytwórni z Dalekiego Wschodu okazała się zbyt silna i firma zaczęła ponosić straty, które w bieżącym roku osiągnęły apogeum. Konieczna okazała się restrukturyzacja przedsiębiorstwa, w ramach której sprzedano prawa do produkcji komputerów PC pod znakiem firmowym Commodore i przeniesiono produkcję Amig z niemieckiego Braunschweigu na Filipiny. Nowemu zarządowi firmy udało się o 70% (!) zmniejszyć koszta własne. Te zdecydowane posunięcia przyniosły już rezultaty i po pięciu kolejnych kwartałach "pod kreską", w trzecim kwartale bieżącego roku osiągnięto zyski. Od początku lipca do końca września Commodore sprzedał w Niemczech ok. 100.000 komputerów, w tym 18000 Amig 1200 i 3000 Amig 4000/040. Do końca września sprzedano także ok. 12000 Amig CD32.

Dynamicznie rozwija się sprzedaż Amigi CD 32. Na rynku jest już dostępnych ok. 160 tytułów na dyskach CD. Przebojowa kampania reklamowa w telewizji niemieckiej i w innych krajach Europy Zachodniej przynosi wyraźne efekty, szczególnie widoczne w Wielkiej Brytanii, gdzie konsola ta

ciąg dalszy na stronie 76



p. Alwin Stumpf; konferencja prasowa Commodere

AMIGOWIEC
ROK 4 NR 9/10 (33-34) INDEX 32034X

REDAKCJA

REDAKTOR NACZELNY:
RYSZARD KOWALSKI

Z-CA REDAKTORA NACZELNEGO:
KRZYSZTOF NOWICKI

OPRACOWANIE GRAFICZNE:
KRZYSZTOF WIRSZYŁŁO
MONIKA BEYGER

SKŁAD KOMPUTEROWY:
MAŁGORZATA LEWANDOWSKA

KOLPORTAŻ:
ANDRZEJ KENTZER

PRENUMERATA:
AGNIESZKA PRZYBYLSKA

SEKRETARZ:
TOMASZ ŁOBODA

WSPÓŁPRACA:
WOJCIECH BIAŁKOWSKI,
WOJCIECH CZYŻ,
JAROSŁAW CHROSTOWSKI,
MARCIN GACKOWSKI,
ADAM GREGROWICZ,
TOMASZ FLANC,
TOMASZ HRYCUNIAK,
MACIEJ KLIMKIEWICZ,
TOMASZ KOKOSZCZYŃSKI,
SEBASTIAN KŁOMSKI,
TOMASZ KULBACKI,
TOMASZ ŁOBODA,
ARTUR ŁUKASIK,
TOMASZ MATAJEK,
IZABELA SKIBIŃSKA,
MAREK STOR,
KRZYSZTOF TROJANOWSKI,
DARIUSZ TRZOS

ADRES REDAKCJI:
ul. Świętojańska 2/7
85-017 Bydgoszcz
tel. (+52) 28-79-20, fax (+52) 22-64-03

KONTO BANK PKO S.A. BYDGOSZCZ
00509011-04009339-2511-30-001110

SKŁAD NA KOMPUTERACH AMIGA
OKŁADKA: WOJCIECH BIAŁKOWSKI
DRUK: P.P.H. "HEKTOR",
STRZELCE GÓRNE1, 86-022 DOBRCZ

Gwiazdka tuż, tuż!



Tak, Drodzy Czytelnicy, to nie pomyłka. Jak już zdążyliście przekonać się (chociażby po okładce) numer 9-10/93 jest numerem świątecznym. Niestety, nie uda się nam wydać numeru 11-12/93 w tym roku kalendarzowym, dlatego też zadecydowaliśmy, że numer następny będzie oznaczony numerem 1/94 i ukaże się w styczniu roku Pańskiego 1994. Oczywiście jest to zabieg czysto techniczny i prenumeratorzy mogą spać spokojnie. Ze względu na poczynione wcześniej obserwacje nastrojów panujących w redakcji, doszliśmy do wniosku, że w przyszłym roku Amigowiec będzie częściej gościł pod Waszymi strzechami. (A ktoś kiedyś mówił, że ustrój totalitarny jest złym sposobem rządzenia.)

Kolejnym negatywem, przed którym nie udało nam się obronić jest zmiana ceny. Postaramy się jednak zrekompensować ją treścią, objętością i jakością druku czasopisma. Jak zwykle nic nie tracą prenumeratorzy, których zmiany ceny pisma niewiele interesują. Z chwilą wydania tego numeru drożeje także prenumerata i wynosi 138.000 zł za 6 wydań oraz 256.000 zł odpowiednio za 12 wydań. W ramach promocji gwiazdkowej wszyscy, którzy zdecydują się zaprenumerować Amigowca do momentu wydania następnego numeru mogą dokonać tego wpłacając 108.000 zł na prenumeratę półroczną lub 216.000 zł na prenumeratę roczną (specjalny blankiet na końcu numeru). Decyduje data stempla pocztowego. Prenumerator zawsze zyskuje.

Przejdźmy do numeru bieżącego. Poprawiona jakość i 80 stron to pierwsze zmiany jakie rzucają się w oczy. Oprócz systematyzacji dotychczas istniejących działów, zrealizowaliśmy nowy pomysł tzn. WARSZTATY. Cóż to właściwie ma być. Otóż, ma być to część czasopisma, w której będziemy starali się przedstawiać praktyczne podejście do różnych problemów. Na razie wyodrębniliśmy osiem grup problemowych mianowcie: AMOS, AREXX, DOS, DTP, DTV, MODEM, MUZYKA, PROGRAMOWANIE. I tutaj prośba do Was, Drodzy Czytelnicy. Kierujcie do nas swoje problemy! Mile widziane listy z dopiskiem np. WARSZTATY lub PISMA PISEMKA. Zachęcamy Was do współtworzenia tych rubryk. WARSZTATY z założenia będą zajmować się tematyką problemową, nie stanowią więc żadnej konkurencji dla powszechnie lubianej rubryki PISMA, PISEMKA dostarczającej odpowiedzi na konkretne pytania. Notabene rubryka ta ostatnio przyżywa mały regres. (Czyżby w tym kraju wszystko wszyscy wiedzieli?) Cytując jednego z kolegów redakcyjnych "Problemem nie są odpowiedzi, lecz pytania", chcę zwrócić uwagę, że każdy z nas uczy się przez całe życie. Kto pyta, nie błądzi. Radzę to przemyśleć. PISMA, PISEMKA mogą mieć równie dobrze znacznie większą objętość.

Zeszła w stan spoczynku TRYBUNA. Brak jakichkolwiek komentarzy dotyczących życia komputerowego zmusił nas do odłożenia tej rubryki do lamusa. Czy na prawdę nic Was nie denerwuje i nie rusza? Nie wierzę.

Prócz tego w tym numerze jeszcze ciepła relacja z targów w Koloni, nowy bezbolesny sposób montażu twardego dysku do Amigi 1200 i wiele innych ciekawych tematów...

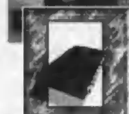
Jako że jest to ostatnie nasze spotkanie w starym roku, życzę Wam w imienu redakcji wszystkiego najlepszego z okazji Świąt Bożego Narodzenia i Nowego Roku. Wielu nieprzespanych nocy spędzonych przed Waszą Amigą z Amigowcem w ręku.

Do siego roku!

R. Kowalski




6 **EMPLANT**
Coś, co może Cię wprawić w zdumienie. MACINTOSH na Amidze. Nigdy nie wątpiłeś chyba w potęgę swojego komputera. Tomek Kulbacki przedstawia rewelacyjną kartę emulacyjną do Amigi.

11 **CD32**
Krzysztof Nowicki przedstawia nowy rewelacyjny kawałek sprzętu. Przekonaj się, że CD32 jest pierwszą tego typu konsolą na świecie.




15 **CED 3.5**
Nowa wersja wspaniałego edytora zagościła na łamach naszego pisma. CED 3.5 to potężny program.




23 **JAK KUPIĆ**
Wojtek Czyż radzi jak zaopatrzyć się w sprzęt oraz jak dobrać go do określonych potrzeb.











Depesze

Tomasz Kokoszczyński

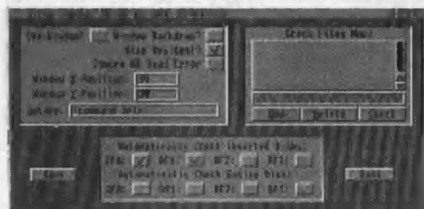
Nowy Virus Checker, czyli... Safe HEX!

Virus Checker jest programem zanym wszystkim poważnym amigowcom. Jeśli chcecie uniknąć szkód wywołanych najnowszymi wirusami powinniście trzymać rękę na pulsie i instalować na dysku coraz to nowsze wersje tego programu. Jest to chyba najlepszy program odwszawiający. Rozprowadzać może go każdy. Rozprowadzać nie znaczy sprzedawać! Tego akurat robić nie wolno. Program został napisany przez Johna Veldthuisa i jest ciągle poprawiany. Nowe wersje ukazują się przeważnie co dwa-trzy miesiące i rozpoznają za każdym razem nowe wirusy.

Autor programu jest członkiem organizacji "SAFE HEX", która zajmuje się walką ze szkodami wyrządzonymi przez wirusy. Przy okazji ścigają "wirusowych" przestępców. Wyznaczyli nawet nagrodę 1000$ za wskazanie z imienia, nazwiska i adresu programisty "od wirusów" (może namówicie jakiegoś kolegę by się zgłosił - może wypłacą). W "normalnych" krajach tworzenie takich programów podlega karze do 5 lat więzienia i tyle go widzieli.

Złapane wirusy można wysyłać na adres:

SAFE HEX INTERNATIONAL
Erik Loevendahl Soerensen
Snaphanevej 10
DK-4720 Praestoe Denmark
Phone: + 45 55 99 25 12
Fax : + 45 55 99 34 98



Co nowego w wersji 6.3? Ostatnio mieliśmy w redakcji wersję 6.2, która została wydana wieki temu. Zmieniło się niesamowicie wiele. Do nowego VirusCheckera wprowadzono wiele nowych opcji, jak np. przeszukiwanie spakowanych plików, korzystanie z biblioteki wirusów bootblockowych. Zresztą spójrzcie na rysunki zamiszczone poniżej, a wszystko stanie się dla Was jasne. Program rozpoznaje w tej chwili następujące wirusy: SCA, AEK, LSD, BYTE BANDIT, REVENGE, BYTE WARRIOR, NORTH STAR/STARFIRE, OBELISK SOFTWORKS CREW, IRQ, PENTAGON CIRCLE, HCS, DISKDOKTORS, LAMER EXTERMINATOR, TIMEBOMB, GADAFFI, BSG9, WAR HAWK, VKILL (or AIDS), ULTRAFOX, PVLPROTECTOR, REVENGE LAMER EXTERMINATOR, UNKNOWN, JITR, MICROSYSTEMS, XENO, 16 BIT CREW, NEW ALIEN BEAT, BLACKFLASH, DIGITAL EMOTIONS, SCARFACE, TURK, JOSHUA, BUTONIC, CENTURIONS, CODERS NIGHTMARE, FORPIB, GX TEAM, GREMLINS, KAUKI, SADDAM virus, CCCP, DISASTER MASTER 2, HAWNES, RETURN OF THE LAMER, TRAVELLING JACK, LIBERATOR, MENEM'S REVENGE, TRABBI, METAMORPHOSIS.

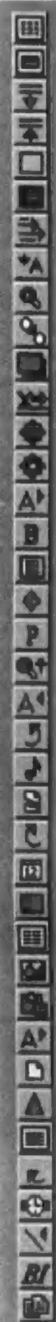
Historia każdego z tych wirusów została dokładnie opisana. Aż się włos jeży jak niektórzy chcą się popisać szerząc głupotę na świat długi i szeroki. Na następnym amigowcowym dysku PD postaramy się zamieścić przynajmniej wersję 6.3 VirusCheckera.



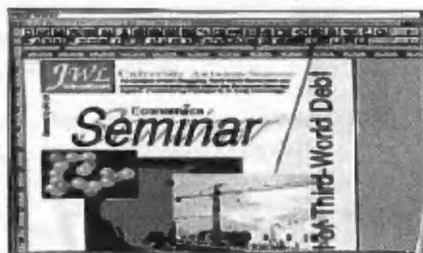
Dobry edytor to połowa sukcesu...

Na Amigę pojawia się coraz więcej profesjonalnych programów. Jednym z nich jest na pewno PageStream 3.0, którego wersja alfa dotarła niedawno do redakcji. Program zapowiada się bombowo, chociaż w tej chwili nie wszystkie opcje działają. Sądząc po stanie programu to jego finalnej wersji możemy spodziewać się za jakieś parę miesięcy. Z pewnością będzie to hucznie opisane na łamach AMIGOWCA.

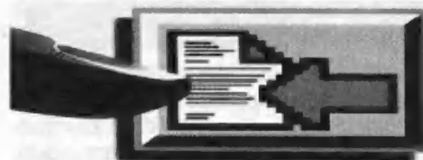
W tej chwili wchodzi na rynek inny, już w pełni działający program. Wszystkim, którzy dokładnie śledzą nasze depesze na pewno nie obca jest nazwa "Final Copy II". Jest to bardzo dobry i bardzo szybki edytor nadający się do różnego rodzaju korespondencji i innych ładnie wyglądających tekstów. Program został wyprodukowany w firmie "Soft Wood". Okazało się, że programiści firmy nie zasypali gruszek w popiele i teraz powstał nowy, lepszy program. Nosi on nazwę "Final Writer" i ma się ponoć bardzo dobrze. Firma (w ramach reklamy) proponuje znalezienie innego programu tego typu na komputery IBM czy też Macintosh, który mógłby tyle samo, a może jeszcze więcej. Chodzi tu o kombinację prostego, ale wyrafinowanego i tak w ogóle najlepszego edytora, z prostym, aż za prostym w użyciu systemem DTP. Podobno w "Final Writerze" to się udało. Tak więc wcale nie musicie mieć obiekcji, że Wasza Amiga może za mało, bo tu ciągle znowu się okazuje, że ona bez problemu może i to bardzo wiele.







Final Writer™



Program jeśli chodzi o tryb edycyjny ma możliwości podobne do "Final Copy'iego II". Oprócz tego został on w pełni dostosowany do potrzeb DTP. Nie tylko można wstawiać przeróżne grafiki do tekstu, ale także opisywać je na tym tekście pod dowolnym kątem. Tworzenie spisu treści, rysunków, czy tabelek ma być po prostu dziecinadą.

Cała obsługa programu jest podobno niesamowicie prosta. Wystarczy wybrać ikonkę i kliknąć na nią. Reszta zrobi się sama.

Jeśli autorom udało się utrzymać szybkość z jaką działa "Final Copy II", to program ten powinien wkrótce stać się Waszym ulubionym narzędziem do tworzenia szkolnych gazetek czy opisów na kasety. Kto wie, czy nie da się na nim tworzyć AMIGOWCA? Gdy tylko "Final Writer" dotrze do nas, postaramy się go dokładniej opisać.

MaxonWord 1.1

Jest to nowy produkt firmy "Maxon". Po długich bólach narodził się wreszcie edytor Maxona. Sądząc po innych produktach, firma jest całkiem niezła. Niezły też powinien być ten edytor. MaxonWord pracuje na zasadzie "kto ciągnie ten jedzie". Na ekranie pokazywane są dowolnie definiowalne strony, tak jak będą wyglądały na wydruku. Na stronach są okienka z tekstami, rysunkami, tak jak w "prawdziwym" DTP. Definiowanie wszelakiego typu obiektów, ładowanie rysunków, czy tekstów odbywa się nie tyle poprzez wyciskanie odpowiednich nazw, co właśnie przez wciąganie odpowiednich piktogramów na odpowiednie miejsce na stronie. Podobnie odbywa się przydział kolorów dokumentom, czy też wykresom.

Program pracuje systemem pudełkowym. Teksty, rysunki i wszelkie inne formy graficzne pakowane są w pudełka odpowiedniej wielkości. Oczywiście to my ustalamy wielkość tych pudełek. Pudełka z tekstem dopasowują się automatycznie do wielkości tekstu, a jeśli jest go za dużo przechodzą na następną stronę. Oczywiście można zmienić łączenie tekstu i ominąć niewygodne przeszkody. Także Rysunki można ładować w odpowiednie ramki. Niestety nie istnieje możliwość wykonywania na rysunkach innych operacji jak np. obracanie.

Strony mogą być oczywiście automatycznie numerowane. Do ustalenia jest liczba kolumn na stronie. Funkcje dzielenia tekstu (niestety bez uwzględnienia wyjątków) pracują w kilku językach: angielskim, niemieckim, włoskim, francuskim. Polskiego jak zwykle (chwała Wam piraci i posłowie) nie uwzględniono. W programie nie ma możliwości tworzenia standardowych listów, a więc dla firm pozostanie on bezużyteczny (pani Asiu, 100 listów z podziękowaniami do 100 najlepszych klientów). Nie ma też kontroli poprawności pisowni. Istnieje jednak połączenie z innymi programami przy pomocy portu ARexxa, co w sumie od biedy umożliwiłoby tworzenie standardowych listów i wypełnianie ich odpowiednimi danymi, ale jakoś nikomu się nie chce tego zrobić. Podsumowując: program jest szybki i prosty w obsłudze. Można go polecić wszystkim początkującym i lubującym się w ikonkach. Być może napisano go specjalnie dla grafików, nie mających czasu na zgłębianie tajemnic menu i wyznających zasadę "co widać, to jest" (WYSIWYG)?

Szparka sekretarka

Czego to ludzie nie wymyślą. Ostatnio okazało się, że naszą przyjaciółkę można przerobić na sekretarkę. W tym celu trzeba posiadać modem ZyXEL i dodatkowo zakupić program "MultiAnswer" firmy Bi-Bu-Soft z Berlina. Program wbrew pozorom nie stwarza problemów przy zainstalowaniu i łatwo daje się konfigurować. W pełni możliwa jest też współpraca programu z programami do obsługi Faxu, tak, że jeśli dociera do nas jakiś fax, to wszystko trafi do odpowiedniej szufladki.

Jeśli teraz ktoś zacznie coś gadać do naszego telefonu, to on nie tylko sam się odbierze, ale rozpozna, czy chodzi o modem, czy też o gadułę i w tym drugim przypadku nagra nam całą gadkę na dysk. Dobrą jakość przy odtwarzaniu zapewnia tryb CELP, który zużywa 1.2 KByta na sekundę. Tryb ten funkcjonuje jednak tylko z niektórymi modemami ZyXELa. Inne tryby zużywają do 3.5 KByta na sekundę, co dla posiadaczy dużych twardych dysków, których nikt nie jest w stanie zapchać, nie jest żadnym problemem. Oczywiście wszystkie dane można też magazynować na zwykłej dyskietce. Oprócz sprawdzenia informacji na odległość można za pomocą tego systemu wystartować dowolny program (chciałbym zobaczyć jak się komuś to całe żelastwo zawiesi).

Gdy powiedziałem kumplowi, że ten programik kosztuje jedyne 150 DM, to on rozsądnie stwierdził: to lepiej chyba kupić sobie sekretarkę. Miał rację.

Struktura to podstawa

Duża część programowania opiera się na tworzeniu algorytmów o czym pisaliśmy swego czasu w amigowcowym kursie "Gdzie Bije Dzwon?". Niektórzy programiści nigdy algorytmów nie tworzą. Ci jednak, którzy ich posmakują, (no w uproszczonej formie) chętnie do nich wracają. Takie algorytmy zostają przetworzone na odpowiednie struktury i stąd wywodzi się nazwa programowanie strukturalne. Programy pisane w ten sposób są jasne i przejrzyste. Tyle tylko, że czasami można się pogubić.

W takich wypadkach pomocą służy programik "Strux". Program ten tworzy z kodów źródłowych napisanych w C lub Pascalu strukturogramy, przebiegi programów, czy też pseudokody. Można w nim również tworzyć gotowe programy, które potem dadzą zapisać się w takich językach, jak Amiga-BASIC, C, Pascal, ARexx. Dodatkowo można definiować własne tabele przekształceń dla innych języków. Gotowe programy dadzą się także bez problemu przedstawić w postaci grafik.

Programy przygotowane przez Strux nie są gotowe do użytku, a stanowią jedynie szkielety właściwych programów. Celem Struxa nie jest zastępowanie edytorów i języków, a rozwijanie myślenia strukturalnego oraz planowanie i rozwijanie programów. Przy okazji używania "Struxa" możemy mieć pewność, że trzymamy się linii postępowania wyznaczonej dla programistów przez Commodore. Program nie współpracuje niestety z assemblerami.

Strux oszczędzi Wam zapewne wiele bezsennych nocy, które dzięki niemu spędzicie w ciepłych i wygodnych łóżkach zamiast ślęcząc przed monitorem. Ze "Struxem" zdrowiej!❑



Emplant

Emulator uniwersalny

Tomasz Kulbacki

Konstruktorzy amigowskich rozszerzeń sprzętowych nabrali ostatnio ciekawego zwyczaju. Mianowicie reklamują towar, którego nie są w stanie dostarczyć, a co więcej: nie wiedzą nawet w przybliżeniu, kiedy znajdzie się on w sprzedaży, mimo, że chodzi o ich własne produkty. Najnowszymi przykładami takiego postępowania są firmy X-pert (karta graficzna Merlin), Village Tronic (karta graficzna Picasso) i wreszcie Blue Moon, z reklamowanym emulatorem Emplant.

W każdym z tych przypadków, producent zamieszczał przez pół roku, w praktycznie wszystkich liczących się w RFN czasopismach amigowskich reklamy, (każda na okładce po 10000 DM za sztukę) z podanymi cenami, adresami i telefonami, pod którymi można było składać zamówienia. Na pytanie o termin dostawy, udzielana była odpowiedź wg. wzoru:

(*) Data dostawy = aktualna data + 14-21 dni

przy czym miała ona znaczenie wyłącznie propagandowe. Po upływie w.w. terminu, kolejny telefon pozwalał ustalić nową datę dostawy - patrz *(*)*.

Sens takiej tego rodzaju postępowania pozostaje dla mnie całkowicie niezrozumiały. Być może jest to nowa strategia reklamowa wyzwalająca nieodpartą potrzebę zakupu sprzętu, kiedy wreszcie znajdzie się on w sprzedaży. W moim przypadku wyzwalała ona tylko zimną furię przy każdej rozmowie z pracownikami w/w firm.

Po mniej więcej 6 miesiącach tej zabawy w ciuciubabkę, sprzęt rzeczywiście zaczął pokazywać się na rynku. Pierwszymi dwoma przykładami zajmę się pewnie w przyszłym miesiącu, a w tym artykule tematem głównym jest Emplant - emulator (jak twierdzi producent) dowolnego komputera. Na razie ta dowolność ogranicza się do Apple Macintosh (w zapowiedzi IBM PC), co już samo w sobie jest warte uwagi.

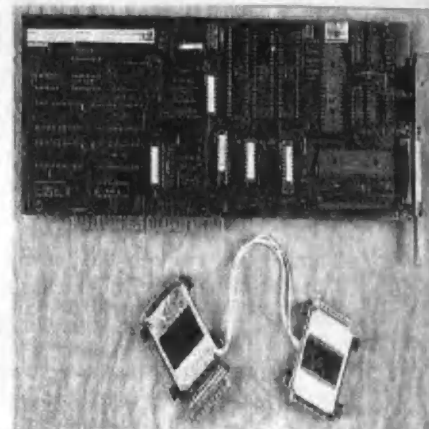
Jedyny dotychczas dostępny emulator Macintosha (AMax Plus) firmy Ready Soft, funkcjonuje wprawdzie tak dobrze, jak tylko emulator może działać, ale emuluje model MAC-Plus o czarno-białej grafice, a w dodatku nie może współpracować z Amigą 4000.

Emplant jest umieszczony na karcie standardu ZORRO II, co implikuje możliwość jego współpracy z Amigą 2000, 3000 i 4000 (040). Posiadacze Amigi 4000/030 nie uruchomią Emplanta - wymaga on procesora wyposażonego w MMU. Funkcjonowanie ze standardową Amigą 2000 jest również niemożliwe, gdyż Emplant wymaga procesora od 68020 w górę. Niezbędny jest więc zakup karty Turbo Commodore A2620 i dodatkowo układu MMU MC 68551, bądź też dowolnej karty turbo z procesorem 68030 lub 68040 (nie Fusion Forty). Instalacja karty i oprogramowania nie stanowi specjalnego problemu, choć pojawia się przy tej okazji pierwszy hak. Emplant nie jest wyposażony w ROM z systemem operacyjnym MAC'a, który trzeba dokupić ekstra (ca 550-600 DM). System można naturalnie skopiować na twardy dysk, co wprawdzie jest nielegalne, ale funkcjonuje równie skutecznie i nic nie kosztuje.

Po instalacji oprogramowanie wychodzi na jaw pierwsza wada Emplanta. Ze względu na ilość pamięci w Amidze (standardowo 6 do 10 MB) Emlant nie działa zbyt porywczo. Pamięci starcza tylko na załadowanie systemu i jednego programu, nie wspominając o pełnym wykorzystaniu możliwości systemu Finder 7.0 (taki Macowy Workbench). Bez zwiększenia ilości pamięci do 18 MB praca z programami DTP lub graficznymi nie jest możliwa.

Grafika

Stopień wierności emulacji grafiki kolorowej zależy od używanego komputera. Z Amigą 2000 i 3000 Emplant może pracować w 16 barwach, z A4000 w 256. Z komputerem wyposażonym w 24-bitową kartę graficzną, można uzyskać nawet 16,8 mln. kolorów, pod warunkiem posiadania odpowiedniego drivera. W tej chwili dostępny jest tylko driver do Retiny i (wkrótce) do Merlina.





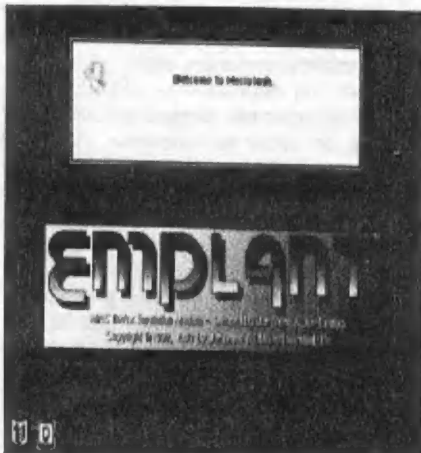
Dostępne są wszystkie możliwe rozdzielczości obrazu komputera, bądź też karty graficznej.

Współpraca z dyskami twardymi i stacjami dysków

Emplant jest wyposażony w zintegrowany kontroler SCSI, mogący pracować z dwoma twardymi dyskami MAC'a. Próby podłączenia dysków z nośnikiem wymiennym (SyQuest 5110) wykazują niestety niechęć współpracy Emplanta (czy raczej kontrolera) do tego urządzenia. Kontroler jest zdecydowanie niedopracowany. Producent zapowiada jednak szybkie usunięcie jego wad i dodanie możliwości jego wykorzystania od strony systemu operacyjnego Amigi. Tymczasem istnieje moliwość nabycia okrojonej wersji Emplanta, pozbawionej kontrolera SCSI. Jest to rozsądne rozwiązanie, gdyż emulator ma dostęp do wszystkich urządzeń Amigi i można wykroić dla niego jedną lub kilka partycji (max. 30 MB) na dysku Amigi. Bezpośrednia wymiana danych pomiędzy Emplantem a Amigą na razie nie jest niestety możliwa. Ten kłopot można obejść stosując kabel Null-Modem, albo dysketki w formacie MS-DOS, odczytywalne zarówno przez Amigę, jak i MAC'a. Najskuteczniejszym rozwiązaniem jest stworzenie na dysku twardym Amigi, partycji MS-DOS, podobnie jak dyskietki nie sprawiającej kłopotu obu systemom. Fakt, że Emplant pracuje w multitaskingu z Amigą sprawia, że ten sposób wymiany danych nie jest specjalnie uciążliwy.

Teoretycznie nie powinno stanowić problemu zapisywanie i odczytywanie dyskietek HD Macintosha (oczywiście pod warunkiem posiadania stacji HD w Amidze), zaś praktycznie, proces zapisywania danych można opisać zmienną losową o rozkładzie zero-jedynkowym, o prawdopodobieństwie porażki niestety sporo większym od prawdopodobieństwa sukcesu. Pewniejszym sposobem jest zapisywanie danych w specjalnym formacie Emplanta, a następnie konwersja dołączonym programem. Producent obiecuje usunięcie tej wady, ale na razie sprawa wygląda j/w Praca z dyskami DD wymaga zakupu dodatkowego urządzenia montowanego na złączu Centronics (o nazwie Sybill).



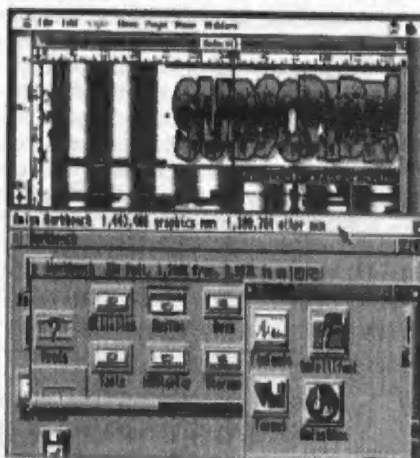


Akcesoria dodatkowe

Oprócz uprzednio wspomnianego kontrolera SCSI, firma Blue Moon, oferuje złącze Apple Talk i interface MIDI.

Kompatybilność i szybkość działania

Kompatybilności Emplanta nie można właściwie nic zarzucić. Wszystkie programy oparte na systemie operacyjnym funkcjonują bez zarzutu. Problemy wykazuje jedynie Microsoft Word, którego autorzy wykorzystali sztuczki programowe normalnie nie stosowane.



Pomiary szybkości funkcjonowania programów wykazują niezwykle dobre rezultaty osiągane przez Amigę z Emplantem. W niektórych przypadkach jest ona szybsza!!! niż emulowany MAC z takim samym procesorem. Amiga 4000 osiąga około 90% szybkości Macintosh Quadra 700. Gorzej wyglądają testy szybkości grafiki, ale tutaj wynik zależy od zastosowanej karty graficznej. Uprzednio wspomniany Macintosh Quadra 700 jest około 1,6 raza szybszy od Amigi 4000 bez karty graficznej. Jak twierdzi producent Emplanta, zastosowanie szybkiej karty graficznej (np. Merlina) pozwoliłoby na uzyskanie lepszych wyników od jakiegokolwiek MAC'a.

Podsumowanie

Emplant jest warty zakupu dla każdego, kto posiada Amigę i potrzebuje Macintosha. Pomimo, że Apple ostatnio sporo obniżył ceny wszystich modeli MAC'a, nabycie Emplanta jest korzystniejsze, nawet jeśli nie bierze się pod uwagę przyszłościowych możliwości jego rozszerzenia (np. o emulację PC). Usunięcie drobnych niedociągnięć, które opisałem powyżej, nie pociągnie za sobą wysokich kosztów, gdyż producent gwarantuje nabywcy darmowy update oprogramowania. Cena Emplanta z Apple Talk, MIDI i kontrolerem SCSI wynosi ok. 1200 DM ❑



Krótki opis procesorów Motoroli od 68000 do 68040

Tomasz Kulbacki

Zainstalowany w pierwszych modelach Amigi procesor 68000 nie jest bynajmniej pierwszym produktem firmy Motorola, jednak ponieważ od tego typu rozpoczyna się związek Motoroli z Amigą, nie widzę potrzeby zajmowania się jego starszymi modelami (6800, 68008).

MC 68000 jest jednostką centralną Amigi 1000, 500, 600 i 2000. Jest to procesor 16-bitowy, o wewnętrznej architekturze 32-bitowej i 24-bitowej szynie danych, pozwalającej na adresowanie ok. 16 MB pamięci RAM i ROM. Dla porównania dodam, że odpowiada on mniej więcej konstrukcją procesorowi 80386SX konkurencyjnej firmy Intel, lecz jest od niego wolniejszy z powodu mniejszej częstotliwości taktowania.

MC 68010, to nieco ulepszona wersja MC68000 stworzona z myślą o oprogramowaniu wymagającym jednostki zarządzania pamięcią MMU (Memory Management Unit). Procesor 68010 jest nieco szybszy od swego poprzednika i może współpracować z układem MMU oznaczonym symbolem MC 68451. Znam maniaków, którzy montowali ten procesor w Amidze 500 w nadziei uzyskania większej szybkości działania, ale jedynym zauważalnym efektem była odmowa współpracy komputera z kilkoma grami (np. Millennium 2.2).

Prawdziwy postęp ustanowił dopiero wypuszczony na rynek procesor MC 68020. Jest to układ w pełni 32-bitowy z 32-bitową szyną adresową, wyposażony w 256 bajtów pamięci Cache. MC 68020 może współpracować z układem MMU typu MC 68551 i koprocesorem matematycznym MC 68881 lub 68882. Procesor Motorola MC 68020 jest montowany seryjnie w Amigach 1200 (w wersji EC o szynie adresowej zubożonej do 24 bitów, bez miejsca na MMU) i w licznych kartach turbo. Niestety, możliwość zainstalowania MMU umożliwia tylko karta Commodore A2620. Procesor ten jest odpowiednikiem Intela 80386 DX, ale podobnie jak w poprzednim przypadku, ze względu na zwykle relatywnie niską częstotliwość taktowania, bywa od niego wolniejszy.

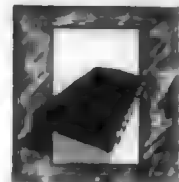
Kolejnym stopniem ewolucji procesora Motoroli, jest ulepszona wersja MC 68020 oznaczona symbolem MC 68030. Dodano w niej drugą pamięć Cache i MMU. Procesor ten przewyższa swymi osiągami 80386DX Intela. Montowany jest seryjnie w A3000 i A4000/030 (w wersji EC - bez MMU) oraz w wielu kartach przyspieszających. Podobnie jak MC 68020 może współpracować z koporcesorami MC 68881/2.

W tym miejscu chciałbym zwrócić uwagę Czytelników na rolę MMU w komputerze. Jednostka zarządzania pamięcią jest niezbędna do funkcjonowania wielu programów i rozszerzeń sprzętowych. Bez MMU nie będzie działać system Unix. Programy zarządzajądze pamięcią wirtualną i niektóre emulatory sprzętowe (np. rewelacyjny emulator Macintosha - Emplant) również odmówią współpracy. Nie będzie możliwe skopiowanie systemu operacyjnego do FAST-RAMu, a co za tym idzie poważne zwiększenie szybkości komputera. Potencjalnym kupcom kart turbo radzę zwrócić uwagę na obecność MMU. Pozornie tania karta turbo może wabić swą ceną ze względu na brak jednostki zarządzającej pamięcią. Różnica ceny MC 68030 i MC 68EC030 wynosi ok. 300-350 DM i porównując ceny kart turbo trzeba mieć to zawsze na uwadze. Jeśli nie możecie bezpośrednio odczytać oznaczenia na obudowie porcesora, sprawdźcie przed kupnem obecność MMU programem obsługującym pamięć wirtualną np. GigaMem niemieckiej firmy BSC. Żadne programy typu Benchmark (SysInfo, AIBB) tutaj nie pomogą, gdyż po prostu kłamią!!! Wykazują one obecność MMU niezależnie od stanu rzeczywistego.

Ostatnim, jak dotychczas, dziełem konstruktorów Motoroli jest MC 68040, który, podobnie jak MC 68030, jest procesorem 32-bitowym, zawierającym dodatkowo zintegrowany koprocesor matematyczny oraz MMU. Koprocesor nie jest w pełni kompatybilny z MC 68882 (brakuje w nim paru rozkazów). Ten problem został rozwiązany w A4000/040 przez dołączenie biblioteki emulującej działanie brakujących poleceń. Komputery wyposażone w ten procesor osiągają 2-3 krotnie większe szybkości obliczeń niż te z MC 68830, i to przy o 50% niższej częstotliwości taktowania. Także MC 68040 ma swoje odmiany "dla ubogich". MC68EC040 nie posiada MMU, a MC68LC040 brakuje z kolei koprocesora matematycznego. (Po bardziej szczegółowe informacje na temat MC68040 zapraszamy do artykułu Tomka Matajka "Procesor 68040 - najszybszy z najszybszych" - przyp. red.)

Potencjalny kupiec Amigi 4000/040 powinien także upewnić się w uprzednio opisany sposób, czy jego przyszły komputer posiada MMU. Pierwsza seria Amig 4000 była wyposażona w odmianę EC właśnie, czego Commodore reklamując A4000/040 jako pierwszy komputer z MC 68040 "zapomniał" zaznaczyć w reklamach. Nasza pierwsza redakcyjna Amiga miała taki właśnie procesor, co zauważyliśmy dopiero instalując bezskutecznie pamięć wirtualną niezbędną do pracy ze skanerem. Ponieważ procesy o zniesławienie mogą okazać się w ostatecznym rozrachunku bardzo kosztowne, nie przytoczę epitetów, którymi obrzucaliśmy szefów i pracowników Commodore.

Kolejnym krokiem ewolucji procesorów serii 68... będzie/jest model MC 68060. Motorola twierdzi, że przyniesie on postęp porównywalny ze skokiem jakościowym pomiędzy 68030 a 68040. Ciekawe, jakie oznaczenie będzie nosiła Amiga wyposażona w ten procesor... ❑



PROCESOR 68040

najszybszy z najszybszych

Tomasz Matajek

Od nowych wersji procesora użytkwnicy oczekują przede wszystkim dwóch rzeczy: większej prędkości i kompatybilności z poprzednimi modelami. Jeśli chodzi o szybkość, to firma Motorola już od długiego czasu jest znana dzięki rodzinie procesorów RISC, M88000. Wraz z pojawieniem się procesora MC68040 osiągnięto podobną prędkość jak w rodzinie procesorów RISC przy jednoczesnym zachowaniu kompatybilności z całą rodziną procesorów MC68000.

W skrócie podstawowe cechy procesora MC68040 przedstawiają się następująco:

- pełna kompatybilność z całą rodziną MC68000
- zawiera główną część koprocesora MC68882
- przy użyciu specjalnego pakietu floating point 100% zgodność z koprocesorem MC68882
- 3,5 do 7 razy większa prędkość obliczeń całkowitoliczbowych w porównaniu z procesorem MC 68030 przy tej samej częstotliwości zegara
- przeciętnie 30 razy większa prędkość operacji zmiennoprzecinkowych w porównaniu z koprocesorem MC 68882
- hardware'owy multiplikator
- podział procesora na siedem niezależnych bloków mogących pracować równolegle
- dodatkowe pipelines w jednostkach przetwarzających dane
- dwa osobne fizyczne caches o wielkości 4 kB każdy
- dwie jednostki zarządzania pamięcią (MMU), umożliwiające jednoczesny dostęp do obydwu caches
- możliwość pracy wieloprocesorowej dzięki zastosowaniu układu snoop-logic.

Szczegółowe omówienie budowy procesora MC68040 wykraczałoby znacznie poza ramy tego artykułu (a także objętość całego "Amigowca"), dlatego też poniżej przedstawione zostaną tylko najważniejsze cechy tego procesora.

Kombinacja mikrokodu i hardware

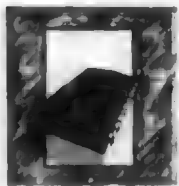
Aż do procesora MC68030 rodzina procesorów 68000 jest mikroprogramowalna. Oznacza to, że wiele z bardziej złożonych rozkazów nie jest wykonywanych bezpośrednio, lecz za pomocą tak zwanego mikrokodu. Mikrokod jest specjalnym wewnętrznym językiem procesora, będącym jeszcze bliżej hardware'u niż assembler, do którego użytkownik nie ma żadnego dostępu. Dzięki mikrokodowi możliwe jest tworzenie bardzo złożonych instrukcji bez potrzeby komplikacji budowy procesora. W ten sposób wykonywane instrukcje potrzebują jednak znacznie więcej czasu do wykonania. Rozwój procesorów RISC podążył w innym kierunku: bezpośrednie wykonywanie instrukcji przez hardware. Zwiększa to co prawda znacznie ich prędkość, lecz jednocześnie ogranicza możliwości i liczbę rozkazów. W procesorze 68040 połączono zalety obydwu filozofii. Jest mikroprogramowalny, lecz mikroprogram wewnątrz wielu skomplikowanych rozkazów został zastąpiony przez odpowiedni hardware. Przykładem tego są wszystkie bardziej złożone tryby adresowania. Do tej pory były one możliwe tylko dzięki zastosowaniu mikrokodu. Przeprowadzono wiele badań i testów, na podstawie których stwierdzono, które tryby adresowania są najczęściej spotykane w praktyce. Dla nich właśnie stworzono specjalny, hardware'owy dekoder zawarty w procesorze MC68040. Również praca obydwu jednostek zarządzania pamięcią (MMU) jest teraz wykonywana przez hardware, a nie przez mikrokod, dzięki czemu szybkość dekodowania adresów drastycznie wzrosła.

Jednostka arytmetyczna

Jak już wcześniej wspomniano, object code dla procesora 68040 jest w pełni zgodny z poprzednimi procesorami tej rodziny. Dzięki wielu optymalizacjom osiągnięto kilkukrotny wzrost prędkości w porównaniu do procesora MC68030 przy tej samej częstotliwości zegara. Na tak duży wzrost prędkości składa się wiele czynników. Jednostka arytmetyczna, w przeciwieństwie do poprzednich procesorów, pracuje z własnym zegarem o częstotliwości dokładnie dwukrotnie większej niż zegar zewnętrzny. Jasne jest więc, że w każdym przypadku procesor MC68040 będzie co najmniej dwukrotnie szybszy niż 68030. Poprzez wiele dalszych optymalizacji (zastąpienie mikrokodu przez hardware, specjalny pipelining) znacznie zredukowano liczbę taktów zegara, potrzebnych do wykonania pojedynczej instrukcji, dzięki czemu osiągnięto wzrost prędkości o kolejny współczynnik 1.8 do 3. Było to możliwe dzięki zastosowaniu specjalnego rodzaju pipeliningu. Mówiąc w dużym uproszczeniu, pipelining jest specjalnym procesem polegającym na pobieraniu kodów rozkazów i ich dekodowaniu jeszcze przed ich wykonaniem. W procesorze MC68040 pipelining jest sześciostopniowy. Każdy stopień wykonuje następujące zadania:

1) Wstępne pobranie kodu rozkazu.
2) Dekodowanie rozkazu.
3) Obliczenie adresu efektywnego.
4) Pobranie danych.
5) Przeprowadzenie obliczeń.
6) Zapisanie wyniku.

Dzięki temu w części arytmetycznej procesora może być wykonywanych do sześciu rozkazów jednocześnie. Jednak w ten sposób skonstruowany pipelining miałby oczywistą wadę: przy każdym wykonanym skoku, podczas przebiegu programu, pipelining musi być wyczyszczony i od nowa załadowany. Ponieważ skoki występują w programie bardzo często (statystycznie



co 5 do 8 instrukcji), bez przedsięwzięcia specjalnych środków efektywność procesów byłaby bardzo ograniczona. W procesorze MC68040 uniknięto jednak tej wady w ten sposób, że dane dla pipeliningu są jak gdyby zdublowane. W danym momencie nie wiadomo, czy skok występujący kilka instrukcji później, będzie wykonany czy też nie. Dlatego też procesor 68040 analizuje dwie możliwości - przy założeniu, że skok zostanie wykonany i że nie zostanie wykonany. Jeśli teraz zostanie wyprowadzony skok warunkowy, to program realizowany jest dalej normalnie. W przeciwnym razie procesor przełącza się automatycznie na drugi pipelining, zawierający już zdekodowane rozkazy. Cały proces został zoptymalizowany dla wykonywalnych skoków warunkowych, ponieważ jak stwierdzono po długich badaniach, realizowanych jest ponad 70% wszystkich skoków warunkowych. Wynika to przede wszystkim z faktu, że skok warunkowy stanowi zazwyczaj kryterium końca pętli i jest zawsze wykonywany poza jednym, ostatnim jej przebiegiem. Bardzo duża wewnętrzna szybkość transmisji danych jest możliwa dzięki temu, że magistrale wewnątrz procesora są przeważnie 128-bitowe. Nowością jest sposób współpracy procesora z magistralą zewnętrzną - procesor teraz nią nie zarządza, lecz tak jak każde inne zewnętrzne urządzenie, musi żądać do niej dostępu. Dzięki temu możliwe jest znacznie efektywniejsze wykorzystanie magistrali przez inne zewnętrzne urządzenia.

Jednostka zmiennoprzecinkowa (FPU)

FPU jest kompatybilna ze standardem IEEE 7541985 i pracuje równolegle z jednostką arytmetyczną. Ma swój własny, trzystopniowy pipelining i własny zestaw ośmiu 80-bitowych rejestrów kompatybilnych z koprocesorem 68882. Trzystopniowy pipelining służy do równoczesnego opracowywania wielu rozkazów i ma następujące funkcje:

1) Konwersja operandów na wewnętrzny, 80-bitowy format.
2) Wyprowadzenie rozkazu.
3) Zapisanie wyników.

Optymalizacja

FPU została zoptymalizowana dla najczęściej używanych rozkazów. Aby osiągnąć maksymalną moc obliczeniową, wbudowano w FPU specjalny, hardware'owy multiplikator. W procesor 68040 wbudowano tylko podstawowe funkcje zmiennoprzecinkowe, ponieważ one właśnie są w praktycznych zastosowaniach najczęściej używane. Dla funkcji trygonometrycznych logarytmicznych i transcendentnych istnieje specjalny pakiet programowy zapewniający 100% kompatybilność z koprocesorem 68882. Jest on związany z procesorem przy pomocy specjalnej metody F-line-traps. Z powodu braku miejsca w artykule nie będę dokładnie opisywał na czym to polega. Ważne jest natomiast, że wszystkie operacje zmiennoprzecinkowe wykonywane są szybciej (nawet te nie optymalizowane), niż przez koprocesor 68882. Przykładem tego jest poniższe zestawienie:

Instrukcja	Liczba taktów zegarowych 68040	Liczba taktów zegarowych 68030/68882
ADD, SUB, CMP (rejestr)	1	2
ADD, SUB, CMP (pamięć)	1	6
SKOK (wykonany)	2	6
SKOK (nie wykonany)	3	4
FADD/FSUB	3	51
FADD.S	3	72
FMUL	5	71
FMUL.D	5	98
FDIV	37,5	103

Jak widać, w niektórych przypadkach różnice są szokujące.

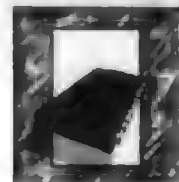
Caches

Budowa obydwu caches w procesorze 68040 jest bardzo złożona i jej dokładne opisanie wykraczałoby poza ramy tego artykułu. Dość powiedzieć, że większa część z 1,2 mln tranzystorów procesora 68040 służy obsłudze i zarządzaniu caches. Postaram się przedstawić tylko główne różnice w stosunku do poprzednich modeli procesorów. Po pierwsze, obydwa caches mają wielkość po 4 kB. Mają one adres fizyczny, a nie logiczny jak w procesorze 68030. Możliwy jest teraz jednoczesny do nich dostęp. Nowością jest także specjalny tryb pracy procesora, zwany copyback. Umożliwia on znacznie lepsze i szybsze wykorzystywanie caches. Fizyczne adresowanie caches ma bardzo poważną zaletę, szczególnie w pracy wielozadaniowej (jak w przypadku Amigi). W przeciwieństwie do logicznych, fizyczne adresy różnych procesów zawsze się różnią i wycinki pamięci z wielu tasków mogą się znajdować jednocześnie w cache. Tak więc przy ich przełączaniu caches nie muszą być czyszczone, a zmienione dane nie muszą być ponownie zapisywane.

Współpraca z pamięcią

Procesor 68040 ma bardzo dużą prędkość, spotykaną dotychczas tylko w niektórych procesorach typu RISC. Jego 25 MHz wersja ma następujące parametry: 15-25 MIPS i 4-8 MFlops (24-40 MIPS i 6.4-12.8 MFlops dla wersji 40MHz). Maksymalne wykorzystanie możliwości procesora nie jest łatwe. Powyższe wahania w jego osiągach wynikają z różnic w konstrukcji hardware'u, w otoczeniu którego pracuje procesor. Osiągnięcie maksymalnych wielkości jest możliwe, ale wiąże się ze zwiększeniem kosztów całej konstrukcji. Niebagatelną rolę odgrywa także szybkość i koszt użytej pamięci. W najprostszym przypadku procesor pracuje w trybie zwanym low drive mode, dla którego wymagana jest teoretycznie pamięć o szybkości 54 ns. Lepszym, choć już nieco droższym rozwiązaniem, jest praca procesora w trybie high drive mode (wymagana szybkość pamięci 45 ns), a najszybszym i najtrudniejszym do praktycznej realizacji jest tryb pracy burst, możliwy do wykorzystania tylko przy użyciu pamięci szybszej niż 35 ns (0 wait states). Dla 40 MHz wersji tego procesora wymagany czas dostępu do pamięci zmniejsza się do 22 ns. Tak szybkie pamięci są niezwykle drogie i dostępne tylko na specjalne zamówienie. W praktyce systemy zbudowane w oparciu o procesor 68040 nigdy nie wykorzystują w pełni jego potencjalnych możliwości, a to głównie ze względu na problemy z odpowiednio szybką pamięcią.

Powyższe zestawienie cech jednostki MC68040 nie zawiera wyczerpujących informacji (na ich objętość wystarczyłaby 300 stronnicowa książka). W treści zawarłem tylko wiadomości podstawowe, które mają służyć ogólnej charakterystyce MC68040. Mam nadzieję, że zagorzali zwolennicy Intela zapoznają się z lekturą tego artykułu i będą bardziej ostrożni w wydawaniu sądów na temat procesorów Motoroli. ❑



AMIGA CD32
druga odsłona

Krzysztof Now[illegible]

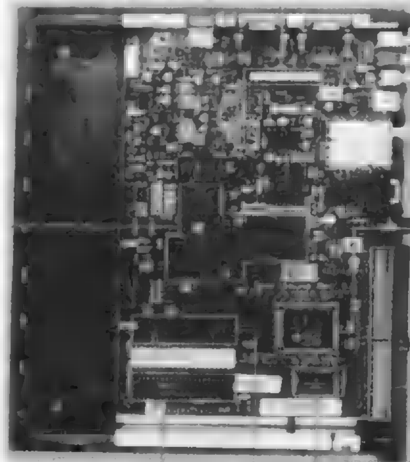
Amiga CD32 - komputer, odtwarzacz kompaktowy, otwarzacza wideo, a może wszystko na raz? Odpowiedź znają z pewnością Czytelnicy posiadający Amigowca sprzed dwóch miesięcy. Powracamy jednak do tego tematu dzięki uprzejmości polskiego przedstawicielstwa Commodore, które udostępniło nam do testów jeden egzemplarz nowej Amigi.

Krótki wstęp

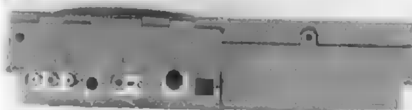
Rozpocznijmy od kilku zdań wprowadzenia, które należą się Czytelnikom nie będącym "w temacie". Amiga CD32 - nowy produkt firmy Commodore przedstawiony w Polsce po raz pierwszy 12 sierpnia (oczywiście 1993 roku), był promowany przez prasę jeszcze przed premierą jako bezpośredni następca CDTV. Dziennikarze mieli racje Amiga CD32 jest następcą CDTV - z kilkoma drobnymi wyjątkami.

Trochę techniki

Nowa Amiga jest wyposażona w procesor 68EC020, układy graficzne AGA, 2MB Chip RAM, czyli jak nie trudno zauważyć jest to Amiga 1200 "wpakowana" w inną obudowę.



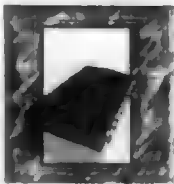
W stosunku do swego poprzednika CD32 różni się kilkoma istotnymi zmianami (oczywiście poza zastosowaniem nowego 32-bitowego procesora i układów graficznych). Nie znajdziemy w niej bowiem złącza szeregowego lub równoległego, nie sposób również odszukać stacji dysków (lub jej złącza), portu PCMCIA, złącza do podłączenia monitora RGB. Jeśli ktoś pokusiłby się o zdjęcie obudowy z pewnością zauważyłby brak złącza do podłączenia dysku twardego. Amiga CD32 nie jest także wyposażona w klawiaturę i myszkę.



Zanim jednak zaczniesz się zastanawiać po co skądinąd poważna firma wypuszcza takie "nie wiadomo co", pozbawione większości niezbędnych do prawidłowej pracy komputera elementów, pomyśl, jakie urządzenie potrzebuje użytkownik, który wykorzystuje komputery jedynie w celach rozrywkowych (czyli tzw. Gracz). Czy jest mu potrzebna klawiatura, drukarka, modem albo drogi monitor? Z pewnością, NIE. On chce się jedynie dobrze bawić. Bawi się więc grami, które muszą:

- posiadać ciekawą oprawę graficzną - co w Amidze CD32 zapewniają układy AGA i wmiarę szybki procesor;
- posiadać interesującą oprawę muzyczną - i dlatego Amiga CD32 została wyposażona w napęd dysków CD;
- być proste w obsłudze - upraszczamy klawiaturę i w ten sposób otrzymujemy specjalny "pulpit" sterowniczy;
- mieć tzw. wodotryski - w postaci np. animowanych sekwencji pomiędzy kolejnymi etapami gry. Tutaj znowu przychodzi z pomocą, dzięki swej pojemności (ponad 600 Kb), dysk CD.

CD i "pulpit" sterowniczy znane są już z pewnością posiadaczom CDTV, uległy jednak "niewielkim" zmianom. Dotyczy to w szczególności napędu CD, który potrafi obecnie przesyłać w ciągu sekundy dwukrotnie więcej informacji (300 kB/s). Natomiast "pulpit" sterowniczy stracił swoją niezależność (w CDTV było to urządzenie działające na podczerwień) i został podłączony do jednostki centralnej kablem (bardzo długim). Projektanci Amigi CD32 doszli także do wniosku, że przeciętny gracz nie jest zainteresowany kupnem monitora RGB (np. 1084) i dlatego zamiast złącza RGB (tzw. szuflady) postanowili wyposażyć ją w złącze telewizyjne. Połączenia z telewizorem możemy dokonać przez złącze antenowe, Composite Video - znane wszystkim posiadaczom magnetowidów oraz złącze S-VHS rzadziej spotykane, ale dające wyższą jakość obrazu. Na zakończenie wypada wspomnieć, że nowy produkt Commodore został "opatrzony" w dwa złącza joysticków (do jednego z nich podłączmy "pulpit" - można także myszkę). Na wszelki wypadek postanowiono także dołożyć złącze klawiatury (identycznej jak w modelach A4000 - niestety



nie ma jej w komplecie) oraz pozostawiono slot bezpośredniego dostępu do procesora (takie jak w Amidze 1200 "pod klapką"). Przeznaczenie tego slotu jest jednak nieco inne. Nie jest ono wykorzystywane do zaimplikowania CD32 dodatkowej pamięci czy karty turbo, lecz do podłączenia modułu MPEG (o którym porozmawiamy za chwilę). W przyszłości firma Commodore planuje wykorzystanie w/w złącza do rozbudowy komputera. Mówi się o całej "skrzynce" zawierającej wszystkie brakujące elementy z Amigi 1200 (złącze stacji dysków, drukarki, modemu, itp.). Aby zakończyć ten krótki opis techniczny jakimś mocniejszym akcentem przedstawiam porównanie "wnętrza" czterech komputerów o podobnym obszarze zastosowań jak CD32. Patrz tabelka strona 14.

Amiga CD32 - komputerem

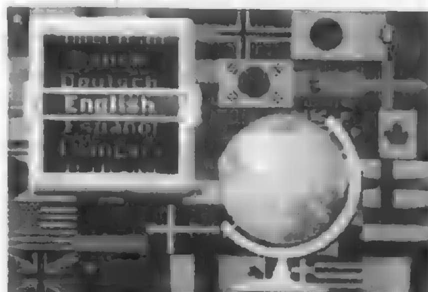
Ponieważ "grzebanie" we wnętrzu Amigi CD32 już zakończyliśmy, trzeba ją trochę wypróbować. Zaglądamy do pudełka, wyjmujemy instrukcję obsługi, otwieramy ją i co widzimy - instrukcja jest niestety po angielsku. W przypadku Amigi CD32 nie jest to co prawda wielkim problemem, gdyż znajdujące się w niej ilustacje dostatecznie jasno tłumaczą sposób instalowania i użytkowania komputera. Na marginesie trzeba jednak zaznaczyć, że przedstawicielstwo Commodore w Polsce mogłoby pomyśleć o dostarczaniu swoich produktów z polskimi instrukcjami. Powróćmy jednak do naszej nowej zabawki. Po udanym podłączeniu wszystkich wystających kabli tj. zasilania, "pulpitu" sterowniczego oraz antenowego do telewizora, uruchamiamy komputer. Po uruchomieniu czeka nas jeszcze dostrojenie telewizora. Operacja ta nie jest trudna, trzeba jednak uważać, aby kabel zasilania i antenowy nie były skrzyżowane (sytuacja taka zdarza się bardzo często ponieważ złącze modulatora oraz zasilania znajdują się moim zdaniem zbyt blisko siebie), gdyż powoduje to potężne zakłócenia obrazu.



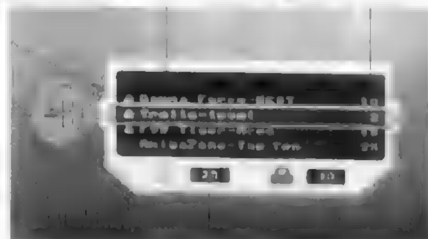
Po prawidłowym ustawieniu telewizora powinniśmy usłyszeć bardzo przyjemne intro dźwiękowe oraz animowany obraz (zamieszczony poniżej).

Teraz wystarczy podnieść osłonę napędu CD i umieścić w nim odpowiedni dysk. Po chwili Amiga CD32 uruchomi napęd i rozpocznie sprawdzanie dysku. W zależności od typu dysku komputer rozpocznie właściwe działanie. Jeżeli typ dysku nie zostanie rozpoznany, jeszcze raz będziemy mogli obejrzeć sobie intro. Załóżmy, że dysk, który umieściliśmy w napędzie, został rozpoznany oraz, że jest to kompakt zawierający program komputerowy. Po takiej identyfikacji program ten zostanie automatycznie uruchomiony. Dalsze postępowanie jest oczywiście uzależnione od tego co zrobimy sami. Najczęściej będziemy mieli do czynienia z grami. Można spokojnie stwierdzić, że na Amidze CD32 będą dostępne tytuły znane posiadaczom Amigi 1200. Jednak "poprawione" przez dodanie np. większej ilości poziomów, animacji oraz wykorzystaniu w oprawie muzycznej dźwięku odczytywanego z kompaktu (16 bitowego).

W testach dysponowaliśmy dwoma dyskami. Pierwszy z nich zawierał cztery demonstracyjne wersje gier oraz także cztery krótkie demonstracje możliwości komputera. Drugi dysk zawierał pełne wersje dwóch gier (Oscar, Diggers - oczywiście na układy AGA). Zabawa była przednia. Na początku trochę popsuły ją nam trudności z obsługą "pulpitu", ponieważ każda gra korzysta zazwyczaj w inny sposób z klawiszy na nim umieszczonych. Podczas tej zabawy odkryliśmy także (ponieważ zgodnie ze zdaniem typowego użytkownika - instrukcja obsługi jest elementem ozdobnym, a więc zbędnym), że Amiga CD32 dysponuje jeszcze dwiema interesującymi funkcjami. Pierwszą z nich jest możliwość zmiany języka, w którym komunikuje się z nami program (o ile oczywiście program taką zmianę umożliwia).



Druga z funkcji udostępnia nam 1 kB "niekasowalnej" pamięci RAM ("nieklasowalnej" to znaczy, że dane w niej umieszczone nie giną po wyłączeniu prądu), w której można przechowywać stan kilku gier.



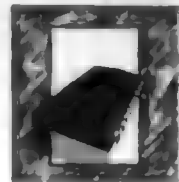
Podsumowując, muszę stwierdzić, że oprogramowanie (szczególnie gry) przeznaczone dla Amigi CD32 jest o wiele bardziej rozbudowane ze względu na rodzaj zastosowanego nośnika od tego, które jest dostępne dla modeli pozbawionych CD-ROM. Umieszczenie programów na dyskach CD ma jednak dla Polaków skutki ujemne. Po pierwsze dyski takie są w naszym kraju bardzo trudne dostępne, a ponadto sporo kosztują. Po drugie zaś, dysku takiego nie można "spiratować" (co wydaje się być najważniejszym powodem małej popularności CD-ROM w ogóle).

Amiga CD32 kompaktem

Powróćmy teraz do momentu, w którym wkładamy dysk do napędu. Może się bowiem okazać, że nie będzie on zawierał programu, lecz muzykę (zwykła płyta kompaktowa audio). W tym przypadku na ekranie telewizora ujrzymy przednią ściankę odtwarzacza kompaktowego, którego przyciski znajdują się na pulpicie sterowniczym.



Nie będę tu opisywał jak się tym kompaktem posługiwać wspomnę tylko, że dysponuje on możliwościami: kolejnego odsłuchiwania utworów, odsłuchiwania utworów w kolejności przypadkowej, odsłuchiwania 10 sekund każdego utworu w kolejności, "przewijania" do przodu i do tyłu. Jakość uzyskiwanego dźwięku odpowiada średniej klasy kompaktom dostępnym w Pol





sce. Amiga CD32 jako kompakt wyróżnia się jedną cechą pozytywną. Posiada możliwość odczytywania nie tylko typowych płyt audio ale również takich, które obok dźwięku zawierają ilustracje graficzne (płyty CD+G) lub wyłącznie podkład muzyczny i plansze z tekstami piosenek (Karaoke). Ponadto możliwe jest także odczytywanie dysków zawierających dane dla instrumentów muzycznych podłączonych poprzez MIDI (którego co prawda nie jeszcze ma) do naszego komputera. Wadą Amigi CD32 pracującej jako odtwarzacz płyt kompaktowych jest brak wyświetlacza, który pozwalałby sterować kompaktem bez potrzeby uruchamiania telewizora (po pewnym czasie można się nauczyć obsługiwania CD32 bez patrzenia na ekran, lecz nie jest to zbyt proste).

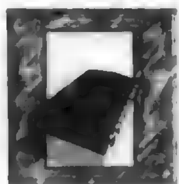
Amiga CD32 odtwarzaczem wideo

Jest to pewne nowum w stosunku do CDTV. Co prawda CDTV dysponowała możliwością odczytywania "filmów" z dysku CD. Posiadała jednak pewne ograniczenia. Były nimi możliwość wyświetlanie obrazu jedynie na fragmencie powierzchni ekranu (nieco ponad 25% powierzchni) oraz ograniczenie ilości kolorów do 4096 (HAM). Amiga CD32 dysponuje możliwością odczytywania "filmów" nagranych na dyskach przeznaczonych dla CDTV (format zapisu nosi nazwę CDXL), a ponadto po umieszczeniu w złączu procesora modułu MPEG, potrafi odczytywać dyski z filmami nagranymi w formacie MPEG. Format ten został pozbawiony wad systemu CDXL. Wyświetlany obraz zajmuje już pełen ekran oraz tworzony jest z wykorzystaniem palety 16.7 miliona kolorów. Filmy zapisane w tym formacie na dyskach CD (do 74 minut) planuje wydawać firma Paramount Home Video. Niestety ponieważ nie otrzymaliśmy modułu, MPEG nie mogliśmy przetestować jak spisuje się CD32 w roli otwarzacza wideo. Na podstawie obserwacji poczynionych w trakcie sierpniowej konferencji prasowej możemy spokojnie stwierdzić, że uzyskany obraz był z pewnością lepszy niż uzyskiwany, w przypadku większości popularnych odtwarzaczy i magnetowidów VHS dostępnych w Polsce. Wątpliwości budzić może jedynie dostępność filmów nagranych na CD w Polsce (oraz ich cena). Nieznana jest nam w tej chwili również dostępność i cena modułu MPEG.

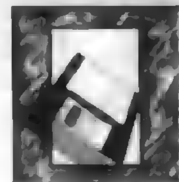
Podsumowanie

Amiga CD32 jest z pewnością komputerem o interesujących możliwościach, stanowiących linię rozwojową CDTV. Komputer ten jest propozycją przeznaczonych dla "nałogowych graczy". Obawiam się jednak, że nie zdobędzie ona dużej popularności w Polsce. Wynika to głównie z niewielkiej dostępności i wysokiej (jak na nasze warunki) cenie oprogramowania, czyli dysków CD. Popularność tego urządzenia obniżą również informacje o opracowaniu napędu CD-ROM dla Amigi 1200. ❑





	3DO	Amiga CD32	MEGA CD	Nintendo CD
Procesor/zegar	ARM60/12MHz	68EC020/14MHz	2*68000/12MHz	65816/3.58MHz
Struktura procesora	32 bitowa	32 bitowa	16 bitowa	8/16 bitowa
MIPS (miliony operacji na sekundę)	6 MIPS	3.5 MIPS	0.3 MIPS	0.1 MIPS
Chip RAM	1 MB VRAM	2 MB DRAM	64 kB VRAM	brak danych
Fast RAM	1 MB DRAM	brak	64 kB SRAM	64 SRAM
Static RAM	Memory Card	1 kB	8 kB	brak danych
Dodatkowe procesory	2 procesory animacyjne, procesor wideo, Digital Signal Procesor, kontroler DMA	Paula, Alice Lisa - obsługa grafiki, dźwięku oraz wejść i wyjść kontrola DMA, przerwań oraz wideo	5 Custom Chipów 2 w Mega Drive 3 w Mega CD	5 Custom Chip w jednostce centralnej i 32 - bitowy koprocesor w napędzie CD
Tryb graficzny	640*400/15kHz	640*512/15 kHz	320*200/15 kHz	512*448/15kHz
Ilość wyświetlanych kolorów	256 z palety 32768	262144 z 16.7 mln.	64 z palety 512	256 z 32768
Szybkość tworzenia obrazu	64 milionów punktów na sekundę	7 milionów punktów na sekundę	brak danych	brak danych
Dźwięk	16 bitowy - stereo lub bezpośredni odczyt z płyt CD, wbudowany Digital Signal Procesor do obróbki dźwięku	8 bitowy - stereo lub bezpośredni odczyt z płyt CD, planowany DSP	[redacted] bitowy - mono lub bezpośredni odczyt z dysku CD	16 bitowy stereo lub bezpośredni odczyt z dysku CD, woudowany DSP
CD-ROM	Zdwojona prędkość odczytu (300 kB/s), MKE	Zdwojona prędkość odczytu (300 kB/s), Chinon/Sony	Pojedyncza prędkość odczytu (150 kB/s), Sony	Podwójna prędkość odczytu (300kB/s) Mitsumi
Wideo (software do odtwarzania)	otwarzanie pełnoekranowe 256 kolorów	tylko część ekranu 4096 kolorów - format CDXL	tylko część ekranu 16 kolorów	nieznana wielkość odtwarzanego ekran 256 kolorów
MPEG	planowany	opcjonalnie	nie	nie
PhotoCD	tak	opcjonalnie	nie	nie
Digitizer	planowany	nie planowany	nie	nie
"Pulpit" sterowniczy	8 klawiszy	11 klawiszy	8 klawiszy	12 klawiszy
Klawiatura/myszka	planowane	opcjonalnie	nie	tylko myszka
Cena	ok. 1800 DM	ok. 699 DM	ok. 900 DM	brak danych
Data premiery rynkowej	koniec 1993	sierpień 1993	wrzesień 1993	koniec 1994



CED
nowa odsłona starego geniusza

Tomasz Kokuszczyński

Jakiś miesiąc temu do redakcji dotarła nowa wersja programu Cygnus Edytor, określanego potocznie jako CED. Ponieważ praktycznie wszyscy w redakcji wiedzą jak się ten program obsługuje, to natychmiast zainstalowaliśmy go w swoich Amigach. A co z tego wynikło historia ta Wam powie...

Nowa wersja nosi numer 3.5 i jest w pełni kompatybilna ze swoimi poprzednikami. Kompatybilna znaczy w tym przypadku, że nie tylko można przenosić teksty z poprzednich CEDów, co jest chyba oczywiste, ale że praktycznie tylko nieznacznie zmieniła się obsługa programu. Ci, którzy przyzwyczaili się do korzystania z CEDa z klawiatury, będą musieli zmienić odrobinę swoje nawyki, ale o tym za chwilę.

Zainsalowanie nowej wersji CEDa polega na przegraniu programu "CED" do katalogu, z którego będziemy go odpalać (czyli w praktyce tam gdzie mieliśmy starego CEDa). Program, można przegrać wraz z ikonką, ale ja zostawiłem sobie starą - kwestia przyzwyczajenia. Oczywiście możliwe jest zainstalowanie programu przy pomocy Installera. Jest to łatwe i wygodne, ale nawet w trybie "Expert" nie daje nam pełnej kontroli nad tym co się dzieje. O ile przy większych programach "Installer" i tak okazuje się niezbędny, o tyle tu możemy sobie dać radę sami. Oprócz przekopiowania samego programu (i ewentualnie nowej ikonki, dla tych którzy lubią ptactwo), należy przekopiować jeszcze kilka plików. Podstawowymi są oczywiście pliki bibliotek wykorzystywanych przez CEDa. Mieszczą się one w szufladzie libs i powinny zostać umieszczone w jej odpowiedniku na dysku, z którego będziemy odpalać CEDa, czyli "Libs:". Tabela I przedstwia biblioteki, z których korzysta CED.

Tabela 1

Directory CygnusED:Libs	
asl13.library *	24988
gadtools13.library	32164
iffparse.library	7688
req.library **	18436

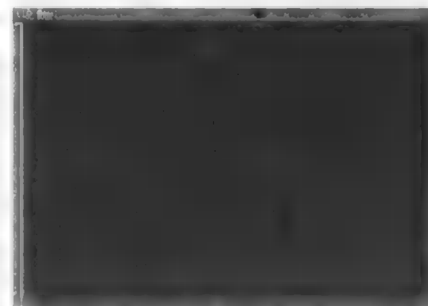
* - biblioteka ze względu na pewne błędy nie jest prawdopodobnie wcale wykorzystana.
** - biblioteka jest wykorzystana tylko w systemie 1.3

UWAGA: W systemie 2.0 i wyżej wprowadzono systemową bibliotekę "gadtools.library", którą w systemie 1.3 zastąpiono specjalną biblioteką "gadtools13.library". Podobne rozwiązanie sprawy biblioteki "asl.library", która jest standardowo dodawana do systemu 2.0 i wyższych, niestety nie udało się. Pozostałe biblioteki nie zostały praktycznie zmienione.

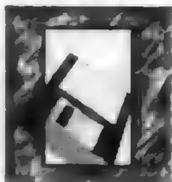
Poza programem głównym i bibliotekami nie musimy już nic więcej kopiować na nasz dysk. Możemy jednak skopiować tzw. "Scripty" - czyli programiki w ARexxie, które pozwalają nam na pełne wykorzystanie programu. Ponieważ ARexx z założenia jest częścią nowych systemów operacyjnych Amigi, to i nowy CED może pełniej korzystać z ARexxa. Tak więc spotkamy wiele programików, których nie można odpalić na poprzednich wersjach CEDa.

Nowy CED 3.5 różni się tylko nieznacznie od swojego porzednika. To nieznacznie oznacza jednak dopasowanie nowego edytora do nowego systemu. Dopasowanie to polega na odnowieniu obsługi programu, poprzez wykorzystanie nowych, systemowych gadgetów i nadaniu oknu programu nieco innego wyglądu. Jeśli chodzi o wygląd to zmiana jest raczej stylistyczna - dodano listwę okna aktualnego tekstu, tak, że jego specyfikacja nie znika przy wybieraniu opcji z menu i wygląda to trochę ładniej.





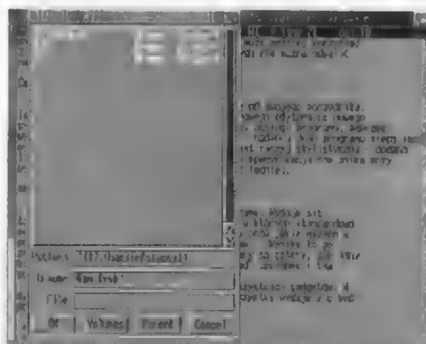
Zmianie uległa również kolorystyka programu. Wydaje się to niemożliwe przy tych czterech kolorach, w których standardowo pracuje CED, a jednak. Ustawienie takich kolorów jakie miałem w poprzedniej wersji wydaje się być niewykonalne... Wynika to ze zmiany przyporządkowania kolorów. Niby kolo



ry są cztery, ale inny kolor odpowiada za listwę i menu, zaś te "od" czcionek i tła zostały niezmienione. Zależy to również od systemu [redacted] jakim się pracuje.

Oczywiście pozmieniano również wygląd wszystkich gadgetów. W zależności od systemu z którym pracujemy wszystko wydaje się być po staremu lub nieco inne.





Gadget wczytywania plików uległ najbardziej istotnej zmianie. Po prostu dostosowano go do systemu. Dla mnie jednak jest to mniej wygodne i to raczej twórcy systemu powinni się dostosować do twórców CEDa, nie zaś na odwrót. Jak wiadomo Commodore swoimi drogami chodzi... Jedyną zaletą tego gadgetu jest to, że przy próbie ponownego wczytania pliku (po jego uprzednim zgraniu), w nowym systemie, mamy zawsze aktualną informację dotyczącą wielkości pliku. W starym systemie musimy uaktualnić ją ręcznie poprzez wybranie opcji "GetDir". W systemach 2.x i 3.x znacznie uproszczono specyfikację plików. Nie ma teraz podziału [redacted] "Show" (pokaż) i "Hide" (ukryj). Mamy po prostu "Pattern", czyli wzór jaki ma być wczytany lub niewczytany. Niewczytanie zapewnia nam znaczek tyldy umieszczony przed nawiasem (~).

Standardowo pojawia się nam coś takiego:

~(#?.(baklautslinfo))

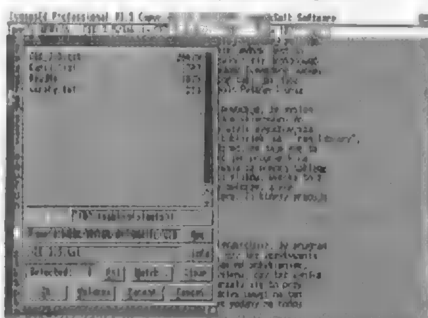
co oznacza, że nie wczytają się pli[redacted] zakończone na:

.bak - ewentualne pliki backupu

.auts - pliki automatycznego backupu

.info - pliki z informacjami o ikonkach

Na całe szczęście jest ktoś taki jak Nico Franćois, który stworzył RTPatch (The ReqTools Patcher) oraz wierni czytelnicy, którzy o mnie pamiętają i dosłali mi ten programik (w tym miejscu specjalne podziękowania dla Kamila). Programik uruchomiony przy starcie systemu powoduje, że system zamiast wykorzystywać gadgety "standardowe" zostaje skierowany do biblioteki "reqtools.library", która jest o wiele wygodniejsza. Zastąpione zostają niektóre funkcje takich bibliotek jak: "req.library", "arp.library" i "asl.library". Muszę powiedzieć, że daje się tu odczuć rękę geniusza, choć kilka rzeczy można by jeszcze poprawić. Postaram się zamieścić ten programik na Amigowym PD. Jedynym mankamentem np. wczytywania pliku za pomocą takiego programiku jest brak uaktualniania jego wielkości i niepodawanie daty. Wynika to z faktu, że dane o plikach przechowywane są w buforze, [redacted] nie uaktualniane przy każdym otwarciu requestera. Ci którzy pracują na dyskietkach wiedzą co to znaczy. Brak daty z niczego nie wynika. Wzory paternów są podobne jak w requestrach systemowych.



A propos wczytywania. Zdarzyło mi się kilkakrotnie, że program zablokował się właśnie w trakcie zgrywania, czy tez wczytywania tekstu. Nie wiem, czy jest to spowodowane jakimś uchybieniem, którego się dopuściłem przy instalowaniu systemu, czy też wynika to z innej przyczyny. Nigdy nie zdarzało się to przy poprzednich wersjach CEDa. Jeśli ktoś ma jakieś uwagi [redacted] ten temat, to prosiłbym o ich przesłanie na adres podany na końcu artykułu.

Skoro już jesteśmy przy nagrywaniu to warto zwrócić uwagę, że pojawił się nowy system zapisu. Teraz nie trzeba potwierdzać automatycznego zapisywania kliknięciem na gadget . Komputer robi to automatycznie, z tym, że tworzy nowy plik z dopiskiem ".auts" (AutoSave). Plik ten jest kasowany w momencie, gdy nagramy nasz tekst. Jest to o tyle wygodne, że mamy do dyspozycji starszą i aktualną wersję tekstu, bez potrzeby bawienia się w ręczne zmienianie nazw. Czasami to się bardzo przydaje... Niestety opcja ta utrudnia życie użytkownikom pracującym na tekstach wczytywanych i zapisywanych na dyskietkach. Po wyjęciu dyskietki w momencie zapisu, program będzie domagał się jej ponownego włożenia w celu skasowania pliku "autosave" (.auts). Uwaga: plik "autosave" zostanie skasowany przy wyjściu z CEDa.

Zmienione zostały wywołania z klawiatury niektórych funkcji dotyczących wycinania. Dla osób, które się przyzwyczaiły do poprzedniego trybu pracy jest to niewygodne tylko na samym początku. Potem okazuje się, że praca na nieszczęsnym IBMie u kolegi staje się łatwiejsza, dzięki temu, że nie mylą nam się przyjęte na całym świecie standardy.

Zmieniono wywołanie funkcji "Insert", która teraz nazywa się "Paste". Kilka razy się zdziwiłem, gdy zamiast wyciętego przed chwilą bloku tekstu pojawiła mi się literka "i", ale wierzcie mi - można się przyzwyczaić. Teraz wstawianie bloku, o przepraszam clipboar

Tabela 2: Stare i nowe wywołania funkcji wycinania

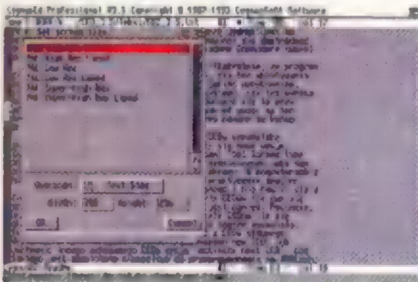
Stare wywołania:	Nowe wywołania:
Block - "A + b"	Block - "A + b"
Columnar block - "A + v"	Columnar block - "A + Shift + b"
Cut - "A + x"	Cut - "A + x"
Insert - "A + i"	Paste - "A + v"
Save block to file - "A + n"	Save clip as... - "A + n"
Delete word - "Alt + Del"	Delete word - "Ctrl + Del"
Undelete word - "Ctrl + Alt + Del"	Undelete word - "Alt + Ctrl + Del"
Bck Spc word - "Alt + BckSpc"	Bck Spc word - "Ctrl + BckSpc"
UnBck Spc word - "Ctrl + Alt + BckSpc"	UnBck Spc word - "Alt + Ctrl + BckSpc"



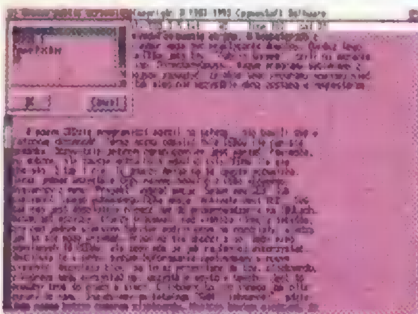
du, odbywa się poprzez wciśnięcie klawisza "Amiga" (określanego jako "Command") i litery "v". Ponieważ przedtem pod tą literą ukryte było wycinanie bloku kolumnowego, teraz zostało ono przeniesione pod "A+Shift+b". Też można się przyzwyczaić. A propos bloku - czy zauważyliście w jaki sposób ominięto to "wyklęte" słowo z menu? Tak jakby nigdy nie istniało. Musiał w tym maczać palce jakiś polityk...

Zmieniono również, tym razem nieznacznie, wywołanie kasowania i wstawiania wyrazu do tyłu i do przodu. Zamiast "Alt" wciskamy "Ctrl" i na odwrót. Patrz tabel II na stronie poprzedniej.

Wracając jeszcze na momencik do ekranu CEDa wypadałoby wspomnieć, że w menu "Enviroments" pojawiły się nowe opcje. Zamiast "Screen height" i "Screen width" mamy "Set Screen type and size...". Wystarczy teraz kliknąć na rozdzielczość jaka nam odpowiada i mamy z głowy przedefiniowanie ekranu. W komputerach z nowymi chipami graficznymi wybór może być praktycznie dowolny.



Oprócz tego możliwa jest opcja otwarcia CEDa jako tzw. "Public Screen", czyli na ekranie dowolnego innego programu (np. DirectoryOpusa). Nazwę programu wybieramy z requestera. Warto w tym miejscu zauważyć, że okno takiego programu powinno mieć wielkość minimum 600*200, tak więc nie wszystkie okna zostaną w requesterze wyświetlone.



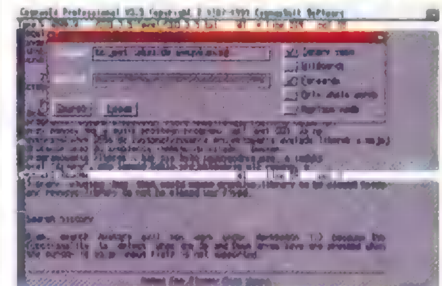
W nowym CEDzie programiści poszli na całego i nie bawili się w fałszywą skromność. Teraz można odpalić tyle CEDów ile nam się spodoba. Oczywiście jedynym ograniczeniem jest pamięć. Powiecie, no dobrze, ale zawsze można było odpalić tyle CEDów ile się chciało. I tak i nie. To znaczy Amiga na to zawsze pozwalała. Teraz jednak wszystkie CEDy możemy odpalić z CEDa głównego. Wystarczy z menu "Projekt" wybrać opcję "Spawn new CED" lub uaktywnić innego schowanego CEDa opcją "Activate next CED". To akurat nie było wcześniej możliwe. Tu widać wyraźnie przewagę Amigi, bo praca z kilkoma edytorami jest absolutnie niemożliwa na IBOLach. A wszyscy, którzy pracowali nad większą ilością tekstów, czy nad jednym większym tekstem podzielonym na rozdziały, wiedzą jak to się może przydać. Przecież nie chodzi o to, żeby mieć odpalonych 10 CEDów, ale żeby móc je jak najlepiej wykorzystać (z jednego drukujemy, na innym piszemy, w trzecim wykonuje się jakiś ARexx). Niestety aktywacja następnego uruchomionego już CEDa (Activate next CED) nie działa u mnie najpoprawniej - tzn. w ogóle.

Pracę z wieloma CEDami bez konieczności tworzenia sztucznych międzyplików umożliwia w pełni system buforowania zastosowany w nowych (2.x i wyżej) wersjach systemu. Wszystkie bloki są teraz przesyłane do tzw. clipboardu, z którego mogą korzystać np. wszystkie edytory tekstu. Jest to poważny krok do pracy w sieci. Clipboard to nic innego jak plik ze specjalnym nagłówkiem zgrany do ramu. Znajdziemy go katalogu "RAM:Clipboards/", gdzie jego nazwa będzie numerem clipboardu. Musicie bowiem wiedzieć, że teraz bez problemu można użyć do 255 takich bloków. CED umożliwia korzystanie dzięki opcji "Set clipboard unit..." z menu "Cut/Paste". Niektóre, starsze programy, w tym poprzednie wersje CEDa pracują na własnych buforach. W starym CEDzie operacje na takich buforach były określane jako operacje na blokach. Teraz zamiast drukować blok - drukujemy clip i podobnie rzecz się ma ze zgrywaniem.

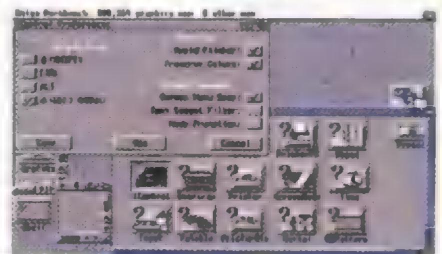
Poprawiono też opcję skoku do "auto-marka". Nie wiem dlaczego, ale w starym CEDzie nigdy mi się nie udało wskoczyć tam gdzie chciałem, o ile ręcznie nie zaznaczyłem sobie tego miejsca. Teraz wystarczy wcisnąć "Amiga + 4" i wszystko odbywa się automatycznie. Ręczne zaznaczanie specyficznych miejsc w tekstach w 3 wariantach (Amiga+1, Amiga+2, Amiga+3) nadal jest aktualne.

Jeśli chodzi o szybkość działania programu to chyba niewiele się zmieniło. Próbowałem porównać obie wersje CEDa i muszę powiedzieć, że np. przy skrolowaniu w obu systemach oba CEDy działają podobnie tzn. nie odczułem większej różnicy.

Zmieniło się też trochę w funkcji wyszukiwania wyrazów, liter, czy czegokolwiek. Nie zmieniły się poszczególne opcje, ale zmianie uległa obsługa gadgetu. Jest on trochę ładniejszy.



W nowych systemach jest możliwa również opcja cofania się do poprzednio wyszukiwanych jednostek. Mówiąc prościej, za pomocą strzałki w gadgecie, w którym wpisujemy wyraz czy tekst do wyszukania, możemy cofnąć się do poprzednich słówek. Na szczęście nie sprawdziła się moja przepowiednia dotycząca niemożności wyszukiwania znaków z "Ctrl". Jest to w pełni możliwe. Zaś aby poszukać w tekście tabulatora należy wprowadzić "Ctrl+I". Aby słowo stało się ciałem należy wybrać w preferencjach "IControl" i wyłączyć tam "Text Gadget Filter". Niby takie proste, ale jak się zapomni jeszcze umieścić w "Startup-Sequence" wywołania programiku "IPrefs", to się można potem zdziwić przy pracy z CEDem.



W nowej wersji dodano również opcję ustalania ilości spacji po znakach przystankowych (kropka, wykrzyknik, znak zapytania, przecinek itp.). Kiedyś próbując formatować tekst opcją "Format - Without fill" otrzymywaliśmy tekst z dwiema spacjami po każdym takim znaku. Teraz przy pomocy



funkcji "Format - Post period spaces..." możemy ustalić ich ilość w zakresie 1-10. Dla mnie bomba, bo lubię mieć jedną spację zamiast dwóch.

Postaram się teraz pokrótce opisać niektóre z nowych scriptów, które dołączono do nowego CEDa. Scripty jak wiecie są to krótkie programiki pisane w ARexxie, które umożliwiają robienie cudów np. przy pomocy tak w sumie prostego programu jaki jest CED. Ja np. wykorzystałem CEDa do zautomatyzowania projektowania wyglądu literek w mojej drukarce oraz do zrobienia indeksu do książki. Owszem programowanie literek i tak nie było najwygodniejsze, a indeks miał "dziury", ale lepsze takie projektowanie niż ręczne, a spróbujcie w ten sposób zrobić indeks na IBMie. Prawdopodobnie tematowi współpracy CEDa (a może również innych programów) z ARexxem poświęcę kiedyś osobny artykuł. Na razie zajmijmy się tym co jest, bez wdawania się w to jak zostało zrobione. Na marginesie dodam jeszcze, że czasami przy próbie współpracy z ARexxem, czy z DOSem nie znikało okno, które normalnie pojawia się w takich przypadkach na ekranie Workbencha. Okno to także od czasu do czasu pojawiało się bez wyraźnej przyczyny. Jeśli ktoś ma jakieś informacje na ten temat to prosiłbym o przesłanie ich na adres podany na końcu artykułu.

Dysponujemy następującymi sciptami:
AltFileError.ced
Calc.ced
CopyToUnit.ced
CutToUnit.ced
DelError.ced
Delete-Until.ced
ExecuteFile.ced
ExecuteLine.ced
JumpToBookmark.ced
List-Dir.ced
PasteFromUnit.ced
Rexx-CED_Demo.ced
Search-For-NULL.ced
SetBookmark.ced
Solve.ced
SortBlock.ced
SpreadViews.ced
Time.ced
ToggleLace.ced
Word-Count-File.ced

Dla przykładu:
"List-Dir.ced" - wczytuje zawartość podanego directory
CopyToUnit.ced
CutToUnit.ced
PasteToUnit.ced - umożliwiają kopiowanie, wycinanie i wstawianie bloku do lub z konkretnych clipboardów.

SortBlock - pozwala na sortowanie wewnątrz zaznaczonego bloku

Oprócz programu głównego i bibliotek na dysku można znaleźć jeszcze kilka innych pożytecznych dodatków.

ED - uaktywnia nowe okno na ekranie CEDa. Można przy jego pomocy z Workbencha wczytać jakiś tekst, do pracującego CEDa.

MetaMac - jest to krótki, ale przydatny programik do edycji poleceń "Macro" CEDa. Przyda się wszystkim tym, którzy przyzwyczaili się do wywołań klawiaturowych różnych funkcji w innych programach (a są tacy?). Przy jego pomocy można tworzyć częściej używane przez nas sekwencje obróbki tekstu.



RecoveryCEDFiles - programik znany wielu z Was z poprzednich wersji CEDa. Przy jego pomocy można bez większego problemu spróbować (he, he) odzyskać tekst po resecie komputera lub po nieopatrznym wyjściu z programu, bez uprzedniego nagrania tekstu. W tym drugim przypadku rzadko odnosi się sukces, tak więc radzę uważać na to co się robi przy wychodzeniu z programu (wiele osób znających angielski nie usiłuje czytać pokazujących się wtedy requesterów!) Mimo wszystko warto mieć ten programik gdzieś na wierzchu...



Ponieważ po moim poprzednim artykule dostałem list z zarzutem, że opisałem tylko program nie podając żadnych konkretnych "trików" (co według mojej opini "lekko" odbieba od rzeczywistości), tak więc teraz postaram się nadrobić to "niedociągnięcie" i pokazać jak łatwiej można z takim programem pracować.

Zacznijmy od wczytywania plików. Jeżeli robimy to z poziomu programu przy użyciu standardowego requestera to nie powinno być z tym żadnych problemów. Oczywiście programy można wczytywać z poziomu CLI (w nowszej odmianie Shella). Tu jednak istotne znaczenie ma przyrostek wczytywanego tekstu. I tak wpisując polecenie:

4.> CED tekst.txt

wczytamy oczywiście żądany tekst, ale przy okazji nowoodpalony CED poszuka w szufladzie "S:" pliku "ceddefaults.txt" (a nie "ceddefaults") i z niego wczyta ustawienie preferencji.

Przy zgrywaniu tekstu na dysk radzę się zastanowić, czy jego poprzednia wersja nie będzie nam potrzebna. Jeśli tak to radzę zgrać tekst pod inną nazwą. Z doświadczenia wiem, że dodawanie cyferek do nazwy nie daje spodziewanych efektów i już następnego dnia można się zdziwić, kto wpadł na tak wspaniały pomysł ponumerowania plików (jeśli jest ich więcej). Radzę wykorzystać pełne możliwości Amigi i nagrywać teksty pod dłuższymi nazwami.

Gdy piszemy list przydatna staje się data. W tym celu zapalamy "CapsLock" (taka zielona lampka po lewej stroni komputera) i wciskamy F10. Pojawi się coś takiego: Wed Sep 29 20:21:36 1993. Wciśnięcie tego samego układu klawiszy, jednak przy przytrzymaniu klawisza "Shift" ukaże nam coś takiego: Sep 29 1993. I tak dalej. Aktualne skróty znajdziecie w tabelce w moim poprzednim artykule (AMIGOWIEC 12/1992 str. 25) (a trzeba było sobie kupić...).

W liście przydatny jest adres nadawcy wypisany zgodnie z najnowszą królującą nam modą u góry kartki i odkreślony od reszty linią. Coś takiego:



Można to uzyskać poprzez wpisanie adresu w linijkach, scentrowanie tych linijek przez kombinację klawiszy Amiga + "", a następnie przez zrobienie jednej lini odstępu i wstawienie podkreślenia przez całą następną linię. Liczbę spacji pomiędzy znaczkami należy wypraktykować doświadczalnie.

ESC[3m ESC[m

Gdy wymieniamy większą ilość różnych "ach" na "och" to niezbędna jest



obsługa requestera "Replace". Możemy wymieniać tekst w trybie globalnie (z podglądem), turbo (bez podglądu, ale szybciej), cofać się do ostatniego zmienionego wyrazu, czy też wymienić pojedynczy wyraz (Yes/No). W razie, gdy wymienimy co nieco za dużo pamiętajmy o funkcji Undo. Funkcja ta niezmiennie podnosi walory programu o 100%. Tak rozbudowanej funkcji Undo nie ma w żadnym programie do obróbki tekstu. Coś takiego pojawiło się w programie "Brillance", ale ten służy raczej do tworzenia grafik niż dłuższych tekstów.

Przy pracy, szczególnie z dłuższymi tekstami, należy używać tzw. markerów, tzn. zaznaczać sobie w tekście miejsca, do których najczęściej skaczemy. Mamy do dyspozycji trzy takie znaczniki uzyskiwane przez kombinację klawiszy: Amiga+Shift+1 lub +2 lub +3. Skok do znaczników odbywa się poprzez Amiga+1 itp. Uwaga: Amiga+4 wywołuje tzw. znacznik autmatyczny, czyli poprzednie miejsce w tekście, z którego gdzieś skoczyliśmy.

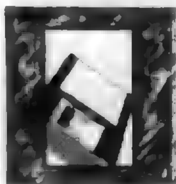
Przy skokach warto nauczyć się przeskakiwania "z klawiatury" na początek/koniec lini/tekstu, czy też skakanie "po ekranach". Jest to niezwykle istotne przy szybkiej edycji tekstu.

Prawdę mówiąc to praca z klawiaturą jest o wiele bardziej wskazana przy programach do obróbki tekstu niż praca z myszą. Dlatego też CED jest taki wspaniale szybki. Odrywanie co chwila ręki od klawiatury by ruszyć coś gdzieś myszą jest mało ekonomiczne. Ci, którzy przyzwyczaili się do poruszania kursora przy pomocy bloku cyfrowego mogą w takim przypadku skorzystać z funkcji "Keypad = movment", która zamienia klawiaturę cyfrową na system IBOLowy. Wspomnijmy w tym momencie minutą ciszy posiadaczy Amig 600, którzy tego szczęścia nie mają...

Jak nauczyć się tych wszystkich skrótów. Nie wiem. Nie trzeba się tu niczego uczyć. Wystarczy od czasu do czasu przy początkach pracy z tekstem zerknąć na skrót podany w menu, a reszta sama wejdzie do głowy. Znajomość podstawowych słówek angielskich w dziedzinie obróbki tekstu można opanować w ten sam sposób.

Tak w ogóle to polecam pisanie 10 palcami i to na dodatek bez patrzenia na klawiaturę. I tu Amiga przychodzi nam w sukurs. Istnieje bowiem programik "Keyboard Kadet", przy pomocy którego, można nauczyć się tego bez problemu. Na początku będzie nam szło o niebo wolniej, ale za to potem o niebo szybciej. A po co się tego uczyć? Logicznie odpowiedź jest prosta: żeby szybciej pisać. Jednak tak naprawdę, to możecie znaleźć sobie inny powód. Od kiedy nauczyłem się tej sztuki ludzie, którzy widzą mnie pracującego przy komputerza mówią: Ale ten to pisze! Nie jest tu dla nich ważne, czy potrafię doskonale obsługiwać 1 czy 50 programów.

Resztę obsługi tego wspaniałego programu pozostawiam Waszej inwencji twórczej. ❑



Vista Professional 3.0

Vista Professional 3.0

Dariusz [illegible]

VistaPro 3.0 jest kolejną ewolucyjną wersją programu służącego do tworzenia krajobrazów. Dzięki zastosowaniu nowych algorytmów (skomplikowana matematyka - fractale 3D) może kreować animacje i krajobrazy o niespotykanym poziomie realności. Niebagatelną rolę, w tej nowej wersji programu, odgrywa specjalny sposób generowania.

Program korzysta ze ztablicowanych wyników pomiarów parametrów świata rzeczywistego i podczas tworzenia obrazu porównuje je z własnymi obliczeniami. Dopiero specjalny moduł programu "artificial intelligence" (sztuczna inteligencja!) wybiera odpowiednie wartości kolorów, tak aby wygenerowany obraz był najbardziej rzeczywisty. Niestety obliczenia są bardzo czasochłonne i pamięciożerne, dlatego VistaPro uruchomi się na komputerach z kickstartem v.2.x wyposażonych w conajmniej 4MB RAM. Jeśli ktoś pragnie wykorzystać tryby AGA musi posiadać minimalnie 6MB. Program rozprowadzany jest w wersji instalacyjnej (HD) na trzech dyskietkach. Oprócz samego programu zestaw zawiera 2 dyskietki z krajobrazami oraz instrukcję.

Użytkownikom, którzy mają problemy z językiem angielskim i obsługą programu, chciałbym na łamach "Amigowca" wyjaśnić najważniejsze funkcje i możliwości programu.

MENU możemy podzielić na 7 części: **Project**, **Load**, **Save**, **GrModes**, **Script**, **ImpExp**, **IQuality**.

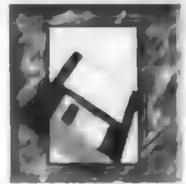
Menu **Project** znajdują się opcje dotyczące rozmiaru krajobrazu, drukowania obrazków, informacje o autorach programu, krajobrazie i wygenerowanym obrazku, a także opcja wyjścia z programu.

Menu **Load** wywołuje procedurę wczytywania, za pomocą której możemy wczytać krajobrazy w formacie: **DEM**, regionu, *USGS*, cyfrowej. Istnieje możliwość załadowania palety kolorów, ustawienia parametrów chmur oraz załadowania tła. Jako tło możemy wstawić obrazek w formacie *IFF* lub *IFF24* (24 bity). Wymagana rozdzielczość 768x484.

Menu **Save** pozwala na zapisanie samego krajobrazu, bądź łącznie z dodatkowymi parametrami (ustawienie położenia kamery celu itp.). Dodatkowo możemy sprawić, aby nasz komputer zgrał na dysk aktualną paletę kolorów, a także ustawienie parametrów chmur. A gdy już się namęczymy i będziemy chcieli zapisać owoc naszej pracy, program umożliwi nam nagranie wygenerowanego obrazku w formacie IFF, IFF24 lub RGB. Dużym udogodnieniem dla posiadaczy Imagina jest z pewnością opcja umożliwiająca konwersję i zapis krajobrazu jako objekt tego programu.

Menu **GrModes** pozwala ingerować użytkownikowi w takie aspekty obrazu jak tryb, ilości kolorów i rozdzielczość ekranu. Następne dwie opcje umożliwiają załadowanie obrazka w formacie *IFF24* oraz umieszczenie go w tle, bądź przed generowanym krajobrazem (wyłączając w generowanym krajobrazie niebo (*Sky*), można załadować inne z dysku). Ostatnia opcja powoduje ciągłe rysowanie krajobrazu w trakcie generowania (wydłuża czas renderingu).





W menu **Script** definujemy ścieżkę po której będzie poruszać się kamera. W tym celu wybieramy pierwszą opcję, która włącza ruch kamery w lini prostej do celu (trzeba wcześniej zdefiniować ilość klatek). Kolejną możliwością jest zapisanie aktualnego skryptu. Edycję następnych klatek dokonujemy przez ustawienie kamery i wybranie opcji *Add. Preview*. Pozwala to obejrzeć tworzony skrypt trójwymiarowo oraz dwuwymiarowo (na mapie topograficznej). *Execute* uruchamia generowanie całego skryptu klatka po klatce. Wyboru formatu zapisywanych ujęć dokonujemy w ostatnich czterech opcjach menu. Opcja *Vanim* umożliwia wybór wewnętrznego formatu VistyPro, który zapisuje plik jako animację. Uwaga! Rozwiązanie to umożliwia korzystanie podczas odtwarzania pliku nie tylko z pamięci, ale również z dysku twadego (pliki mogą mieć rozmiar nawet do 800 MB!). Trzeba jednocześnie pamiętać, że format ten zajmuje dwukrotnie większą objętość niż normalny plik animacyjny (np. w formacie Anim5) i spowalnia prędkość odtwarzania animacji (przy bordzo dużym pliku nawet do 2 klatek na sekundę!).



Menu **ImpExp** umożliwia operacje na palecie kolorów. Można ją przekonwertować na zwykły plik *IFF* (*Col --> IFF*) i na nim rozmieścić drogi, drzewa czy budowle. Później trzeba plik przekonwertować w odwrotną stronę (*IFF --> COL*). Jeśli ktoś chciałby przenieść mapę terenu na plik *IFF*, może skorzystać z opcji *Alt --> IFF*. Konwersja odwrotna (*IFF --> ALT*) może zostać wykorzystana do opracowania krajobrazów na podstawie zdigitalizowanych fragmentów map! Ostatnie trzy opcje służą do skalowania obrazka.

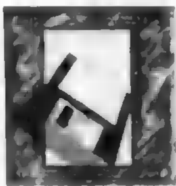
Menu **IQuality** decyduje o jakości generowanych obrazków. Wybierając opcję *Low* decydujemy się na najszybszy rodzaj generowania, lecz prędkość okupujemy jakością. *Medium* pozwala na uzyskanie średnich efektów przy średnim czasie generowania. *High* umożliwia otrzymanie obrazu z dużą dokładnością przy znacznym nakładzie czasu. Zaś tryb *Ultra* pozwala na uzyskanie zaskakujących efektów, ale czas potrzebny na ich uzyskanie liczymy nie w minutach lecz w godzinach! (Amidze 4000/040, przy niezbyt skomplikowanych projektach, zajmuje to kilkanaście minut).

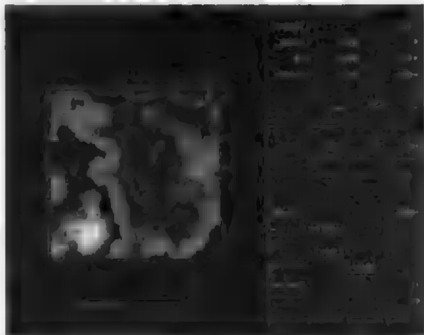


Ekran roboczy programu, oprócz rozwijalnych menu, składa się z paneów kontrolnych (po prawej stronie ekranu), mapy topograficznej (strona lewa) oraz ekranu koordynatów zawierającego współrzędne wybranego objektu w przestrzeni 3D (pod mapą).

Panel kontrolny umożliwia kontrolę nad działaniem programu i zawiera trzy części. Panele **Upper** - górny, **Middle** - średni i **Lower** - dolny.

Panel **Upper**. Tutaj umieszczone są informacje odnośnie położenia kamery i celu (w trzech wymiarach: osie x, y, z), a także odległości pomiędzy tymi objektami (dx, dy, dz, dr - odległość w lini prostej). Gadżet *P* umożliwia ręczne ustawienie kamery w rzucie perspektywicznym. Ustawienie kąta przechyłu kamery możemy korygować gadżetami *Bank*, *Head*, *Pitch*. *Range* pozwala na generowanie krajobrazu jedynie w wyznaczonym promieniu. Dzięki tej opcji tereny położone poza okręgiem wyznaczonym przez promień nie są generowane (oszczędzamy czas!).

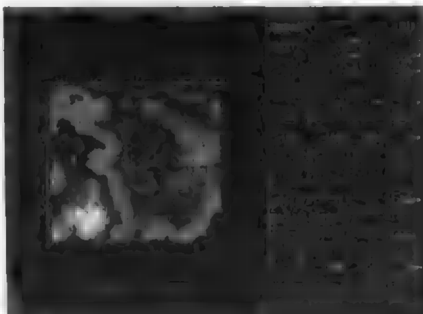




Za pomocą Panelu **Middle** można kontrolować poziom wody (*SeaLvl*), drzew (*TreeLn*), śniegu (*SnowLn*) oraz gęstość mgły (*HazeDn*) w tworzonym krajobrazie. Dzięki takim gadżetom jak **Lake** możemy dodawać do krajobrazu jeziora (*Lake*), rzeki (*River*), gwiazdy (*Stars*), niebo (*Sky*), horyzont (*Horiz*). Możemy także "sadzić" drzewa (*Tree*), budować drogi (*Road*) oraz stawiać budowle (*Bldgs*). Dodatkowe efekty to: tworzenie dolin (*Valley*) i skalnych urwisk (*Clifs*). Posiadacze okularów 3D mogą, za pomocą opcji **Stereo**, wygenerować sobie dwa odrębne obrazki, które po złożeniu na odpowiednim programie, będą symulować trójwymiarowy świat. Do wygładzenia terenu służy gadżet *Smooth*, a do marszczenia *Shrink*. Jeśli ktoś chce powiększyć tylko część krajobrazu powinien wcisnąć gadżet *Enlarg*. Natomiast opcja *VScale* spowoduje powiększenie całego krajobrazu w pionie. Program podczas generacji obrazka za każdym razem zmienia jego paletę. Jeżeli ktoś tworzy animację składającą się z wielu klatek będzie zmuszony do korzystania z jednej palety. W tym momencie pomocna jest opcja *LockP*, która powoduje niezmienność palety w kolejnych klatkach. Oczywiście kolory na ekranie można zmieniać uruchamiając opcję *CMap*.

Najniższy, **Lower** panel jest zbudowany z czterech mniejszych: **Main**, **Lens**, **Frac**, **Light**.

W panelu **Main** można m.in. wybrać ilość wielokątów, z których będzie składać się nasz krajobraz (*Poly* 1 najwięcej *Poly* 8 najmniej). *Dither* powoduje wymieszanie sąsiednich kolorów w celu zwiększenia ich ilości. Gadżet *Texture* (jakość: O - wyłączona, L - niska, M - średnia, H - wysoka) powoduje cieniowanie części terenu położonych najbliżej kamery. Zaś poniżej niego znajdują się trzy opcje, których włączenie zmniejsza ilość czasu potrzebnego na wygenerowanie krajobrazu. *Bound* wyłącza cały teren nie zaznaczony, *BFCull* tylko tę część terenu nie widzianą przez kamerę, a *Blend* redukuje działanie opcji *Texture* na dalszych obszarach krajobrazu. *GShade* powoduje wykonanie miękkich przejść między kolorami.



W panelu **Lens** dysponujemy takimi opcjami jak ustawienie 90 stopniowego kąta patrzenia *Wide*, przybliżanie kamery do celu *Zoom*, ustawienie długości ogniskowej *Fcl Ln* oraz ręczną zmianę szerokości kąta soczewki kamery *Fld OV*. Pozostałe trzy opcje wiążą się z uzyskaniem panoramicznego widoku na krajobraz.

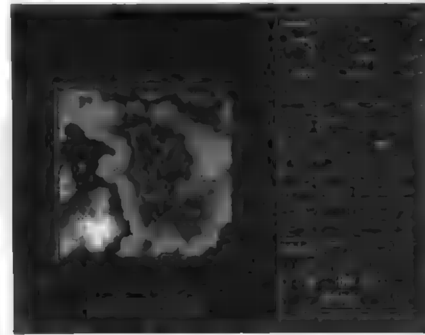
Dzięki panelowi **Fractal** użytkownik może losowo wygenerować krajobraz (*Random*) lub ręcznie wprowadzić numer krajobrazu w polu znajdującym się pod tym gadżetem. Generowany fractal może być wyspą na morzu (opcja *Island*). *FrDim* ustala wysokość wzniesień, *Frctlz* ustala wygładzenie jego powierzchni (1 - gładka, 8 - nierówna). Dodatkowym sposobem na uzyskanie nietypowych efektów jest gadżet *Strtch*. Powoduje on wzmocnienie różnic wysokości tzn. wgłębienie dolin lub podwyższenie szczytów.

Do tego, aby w pełni podziwiać naszą krainę konieczne jest ustawienie światła, służy to tego panel **LIGHT**. Kierunek padania promieni słonecznych ustalamy w nim gadżetami N, S, E, W. Czarne numerowane okręgi umieszczone na mapie topograficznej przedstawiają kąt pod jakim słońce oświetla teren. *Azimth* to kierunek padania światła mierzony pomiędzy południem, a dowolnym kierunkiem geograficznym. *Shadow* pozwala na stworzenie cieni.

Poniżej panelu **Light** znajdują się 4 gadżety umożliwiające sprawowanie bezpośredniej kontroli nad procesem renderingu. **Render** początkuje generowanie, a **Abort** je przerywa. **View** wyświetla wygenerowany obrazek, **Redraw** odświeża **View Screen**.

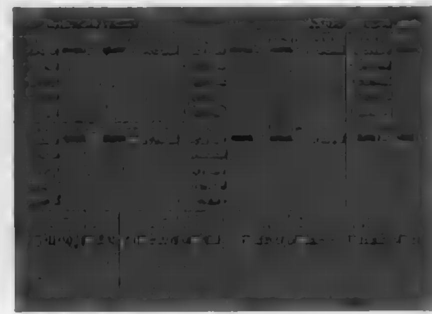
Ukryte klawisze.

Pomimo dokładnego przestudiowania instrukcji programu, użytkownik VistyPro nie znajdzie tam opisu do poszczególnych klawiszy. Dopiero metodą prób i błędów można odkryć, że:



F1,V - wyświetla wygenerowany ostatnio obrazek
F2 - uaktywnia Control Panel
F3 - zapisuje obrazek w formacie IFF24
F6 - przemieszcza mapę terenu
F7 - rozpoczyna wyżłabianie
F8 - odświeża mapę
F9 - włącza lub wyłącza tryb HAM8
F10 - odświerza cały ekran
1,2,4,8, - powoduje zmianę ilości wielokątów
R - generuje krajobraz
T - ustawia cel, tam gdzie znajduje się wskaźnik myszy
C - ustawia kamerę, tam gdzie znajduje się wskaźnik myszy
A - zapisuje aktualną pozycję kamery i celu w pliku VistaPro.prefs
F - losowo generuje krajobraz
Left Amiga i n - przejście do ekranu Workbencha
Left Amiga i m - przejście do ekranu VistyPro

I tak dotarliśmy do końca artykułu. Starałem się w nim przybliżyć Wam magiczny świat krajobrazów komputerowych. Oczywiście informacje jakie znaleźliście w artykule nie zagwarantują Wam natychmiastowego opanowania programu, lecz pozwolą lepiej się w nim orientować. Jak się niedawno dowiedziałem, redakcja planuje wprowadzenie do Amigowca nowej rubryki, która będzie zajmowała się grafiką w praktyce. Może tam znajdziecie bardziej praktyczne porady. (Spójrzcie wgłąb numeru. - przyp. red.) ❑





Jak kupić?

Wojciech [illegible]

Jako że zbliżają się Święta postanowiłem wyjść na przeciw tym, którzy planują w najbliższym czasie sprawienie sobie gwiazdkowej niespodzianki. Być może widzieliście już opisy poszczególnych modeli - jednak nie [redacted] formie porównania modeli CDTV, A500, A500, A600, A1200. Komputery te są bardzo zbliżone cenowo (znacznie droższa jest jedynie A1200), posiadając dość zróżnicowane możliwości. Powyższych modeli nie opisuję odrębnie, lecz porównuję według kolejnych atrybutów.

System operacyjny.

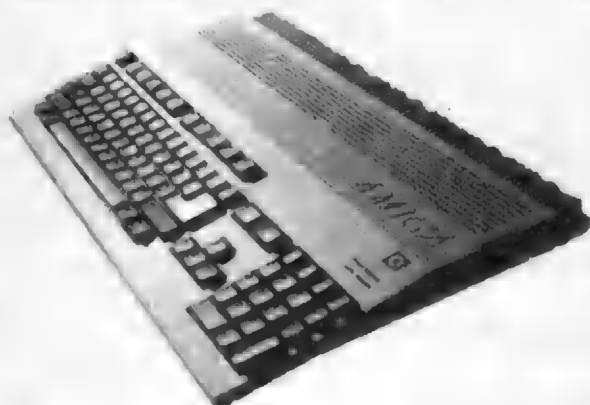
W Amigach system jest zawarty w dwóch miejscach. Najważniejsza jego część (jądro) zawarta jest w tzw. Kickstarcie. Jest to pamięć ROM (niezapisywalna) wbudowana we wnętrze naszego komputerka. Resztę systemu [redacted] zbiór przydatnych programów otrzymujemy [redacted] dyskach. Nowsza wersja systemu operacyjnego to szybsza i wygodniejsza praca, wreszcie rzecz najważniejsza - kompatybilność z określonymi programami. Bardzo stare programy pisane były niezbyt poprawnie - błędy dopuszczalne [redacted] starym systemie blokują nowszy system operacyjny. Commodore zawsze ustalało reguły używania systemu i wykorzystywania coraz to nowszych możliwości Amig. Stąd system operacyjny jest zawsze w 100% kompatybilny (zgodny) ze starszymi wersjami, można raczej mówić o niekompatybilności niektórych programów - wykorzystującymi zasoby komputera niezgodnie z ustalonymi regułami. Liczba programów kompatybilnych jedynie [redacted] najstarszą wersją systemu ciągle się zmniejsza, natomiast coraz więcej programów wymaga minimum wersji 2.0.

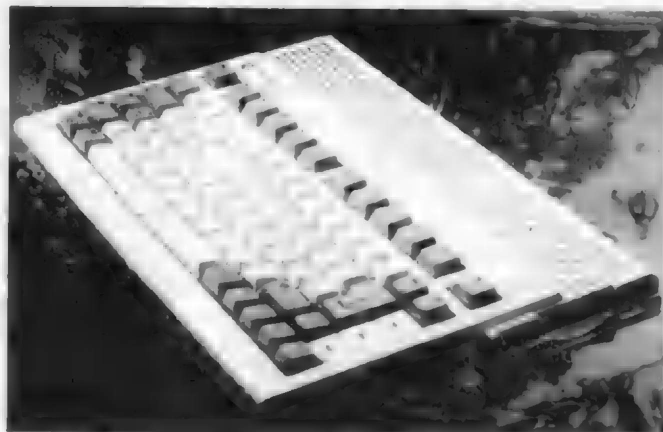
Opiszę teraz systemy używane w opisywanych Amigach

System 1.2 - Najstarszy używany system operacyjny, można go spotkać w bardzo starych Amigach 500.

System 1.3 - powszechnie używnany w komputerach A500 oraz CDTV. Od wersji 1.2 różni go jedynie automatyczne wykrywanie urządzeń zewnętrznych np. twardych dysków. System 1.2 mógł z nich korzystać, jednak bez możliwości startu systemu z takich urządzeń - posiadając twardy dysk trzeba było zawsze wkładać dyskietkę startową aby system z niej wystartował i "zauważył" HD.

System 2.04 - Poprawiona wersja systemu 2.0 wykorzystywana w komputerach A500+ i A600. Wprowadzenie systemu 2.0 było dużym przełomem w Amigowym świecie. Jest on znacznie szybszy i wygodniejszy od poprzednich wersji systemu operacyjnego. Wprowadzono trójwymiarowe obiekty graficzne poprawiające estetykę systemu oraz ułatwiające odróżnianie aktywnych (wypukłych) elementów od części informacyjnej i dekoracyjnej. Jeszcze większe zmiany wprowadzono wewnątrz systemu - dodano dziesiątki nowych funkcji i struktur systemowych, nowe biblioteki porządkujące wygląd programów. Wreszcie wbudowano interpreter języka Arexx (zdefiniowanego przez firmę







IBM, niestety PC okazało się zbyt prymitywne do jego implementacji) łączącego programy w większe pakiety oraz wykonującego powtarzające się czynności operatora. Wprowadzono pełną współpracę ze skalowalnymi czcionkami, czyli powiększanymi lub zmniejszanymi do dowolnego rozmiaru bez utraty jakości.

Dość duża ilość gier i słabszych programów miała początkowo kłopoty z pracą na nowym systemie. W chwili upowszechnienia Amigi 500+ i 600 sytuacja ta uległa radykalnej zmianie. Wszystkie nowe programy współpracują z tymi modelami, a większość nowych aplikacji wręcz go wymaga. Jeśli zamierzasz trochę poważniej lub profesjonalnie wykorzystywać swój komputer, wybierz ten wyposażony w system 2.0 lub lepiej w 3.0.

System 2.05 - Wersja systemu 2.04 sprzedawana z komputerami A600. Różni się jedynie dwoma cechami - W ROMie dodano procedury obsługujące złącze PCMCIA (patrz dalsza część artykułu), a do oprogramowania dyskowego dodano programik konfigurujący dyski krzemowe. Z nieznanych powodów usunięto programy obsługujące syntezę mowy (program Say oraz pliki tworzące urządzenie logiczne Speak:). Programów tych nie ma w systemie 3.0.

System 2.1 - Zubożona wersja systemu operacyjnego 3.0 przeznaczona dla użytkowników komputerów nie wyposażonych w nowe układy graficzne (AGA) i kickstart w wersji powyżej 39. Użytkownik otrzymuje wszystkie udogodnienia systemu 3.0 (bez możliwości wynikających z zastosowania układów AGA, rzecz jasna).

System 3.0 - System zawarty w ROMie komputera (znajdujący się na płycie komputera) zmienił się niewiele. Przede wszystkim poprawiono błędy znalezione w systemie 2.04 oraz dodano funkcje obsługujące nowe tryby graficzne AGA. ROM 3.0 znajduje się jedynie w komputerach wyposażone w nowe układy graficzne czyli A1200 i A4000. Kolejna nowość to pełne wykorzystanie stacji poczwórnej gęstości popularnie nazywanych HD (1.76 MB). Bardzo dużej zmianie uległa część "dyskietkowa". Programy systemowe uzyskały identyczny wygląd, dodano wygodną obsługę procesu formatowania dyskietek. Wprowadzono nowy format który przyspiesza czytanie katalogu dyskietki oraz operacje odczytu.

Do systemu dodano program Cross-Dos obsługujący dyskietki w paskudnym aczkolwiek przydatnym formacie IBM. Wszystkie programy czytają dyski 720KB lub 1.44 KB jak zwyczajne amigowskie dyskietki. Tak więc system 3.0 jest warty prawie 2 mln zł więcej w porównaniu do systemu 1.3 - jest to wartość ARexxa i CrossDosa, które przedtem były sprzedawane oddzielnie.

W systemie usprawniono możliwość zmiany konfiguracji - dodano możliwość lokalizacji, czyli zmiany języka z którego korzystają wszelkie systemowe programy i komunikaty (nowe programy również potrafią zmieniać język w zależności od ustalonego na danym komputerze). W programie ustalającym wzorek, którym system wypełni okna oraz ekrany, dodano możliwość zastąpienia wzorka rysunkiem. Do ustawienia trybu pracy drukarki dodano program współpracujący z PostScriptem.

Amigi posiadają specjalny system oznaczeń wersji i wydania (Ang. Version & Revision). Wydanie dowolnego systemu lub jego części (np. biblioteki) można odczytać komendą Version:

```
1.Work:> Version
Kickstart 39.106, Workbench 39.29

1.Work:> Version Intuition.library
intuition.library 39.2084

1.Work:> Version Scsi.device
scsi.device 37.64
```

Oto lista kolejnych wydań systemu:
30 - Pierwszy system operacyjny V1.0 (A1000)
31 - Wersja PAL systemu 1.1 (A1000)
32 - Wersja NTSC systemu 1.1 (A1000)
33 - System 1.2 (A500, A1000)
34 - System 1.3 (A500, A1000)
35 - System 1.3 plus moduły obsługujące monitor A2024 (A500, A2000)
36 - Próbna wersja systemu 2.0 sprzedawana wraz z pierwszymi modelami A3000
37 - Pełna wersja systemu 2.0 oznaczona 2.04 (A500+, A600, A3000)
38 - Próbna wersja systemu 3.0 (A4000)
39 - Pełna wersja systemu 3.0 (A1200, A4000)

Jeśli chodzi o rozważane modele Amigi, to mają one poniższe systemy:
A500 - System 1.3 lub 1.33
CDTV - Mutacja systemu 1.33, czasem występują niezgodności (nie z winy producenta lecz krnąbrnych programistów ignorujących "zasady zdrowego rozsądku")
A500+ - System 2.04
A600 - System 2.04 lub 2.05
A1200 - System 3.0

Urządzenia zewnętrzne

Stacja dysków elastycznych - Dla wszystkich modeli Amigi jest to stacyjka 3.5" o pojemności 880KB (DD). W przypadku korzystania z systemu 1.3 jej pojemność spada do 837KB. System 2.0 pozwala na pełne wykorzystanie FastFileSystem czyli nowego sposobu zapisu informacji na dysku. Jest to sposób znacznie szybszy i wydajniejszy - pojemność dysku wzrasta do pełnych 880KB. System 3.0 wprowadza Directory Cache, który jeszcze bardziej przyspiesza działanie dysku. Pamiętaj jednak, że stosowanie nowszego systemu zapisu uniemożliwia odczytanie dysku na starszym komputerze. Optymalnym systemem jest system FFS, rozumie go większość programów typu DiskDoctor, po pewnych zabiegach jest odczytywalny na wszystkich komputerach.

Tabela 1

Komputer	CDTV	A500	A500+	A600	A1200
Stacja dysków 3,5"	Nie*	Tak	Tak	Tak	Tak
CD-ROM	Tak	Nie*	Nie*	Nie*	Nie*
Dysk Twardy	Nie	Nie*	Nie*	Tak [1]	Tak [1]
Klawiatura + Mysz	Nie*	Tak	Tak	Tak	Tak
Pilot	Tak	Nie	Nie	Nie	Nie
Złącze PCMCIA	Nie	Nie	Nie	Tak	Tak
Zegar czasu rzecz.	Tak	Nie*	Tak	Nie*	Tak [2]

* - Zawsze można dobrać konfigurację, która będzie posiadała dane urządzenia lub dokupić je później.

[1] - W komputerach tych standardowo zainstalowany jest HD, jednak jest możliwość zakupu bez tego dysku.

[2] - Zależy od konfiguracji.



Tabela 2

Komputer	CDTV	A500	A500+	A600	A1200
Std. Chip	1MB	0,5MB	1MB	1MB	2MB
Max. Chip	1MB w	1MB *w	2MB w	2MB w	2MB
Slow	/	0,5MB w	/	/	/
Fast		8MB z	8MB z	/	4MB w
PCMCIA	/	/	/	4MB z	4MB z

* - Pamięć slow nie może być instalowana jednocześnie z rozszerzeniem pamięci Chip. Fizycznie jest to ten [redacted] układ, tryb działania rozszerzenia wybieramy przełącznikiem na płycie głównej.

z - Pamięć jest instalowana na zewnątrz komputera.

w - Rozszerzenie jest instalowane wewnątrz obudowy.

Dysk optyczny CDROM - Odznacza się ogromną pojemnością ok. 600MB (większą niż typowe twarde dyski). Niestety nie ma możliwości zapisu na taki nośnik, jest więc przydatny jedynie do odczytu fabrycznie zapisanych danych. Znakomicie przydaje sie grafikom - dość popularne są kolekcje gotowych wzorków i rysunków do natychmiastowego użycia. Jest również przydatny w różnego rodzaju prezentacjach multimedialnych oraz dla zapalonych graczy. Typowe gry mają jakość filmów połączoną z 16-bitową muzyką CD. Dysk CDROM posiada jedynie Amiga CDTV, pozostałe modele mają go jako wyposażenie dodatkowe. Jest on podłączany złączem DMA (lewa strona komputerów 500 i 500+), złączem PCMCIA (lewa strona komputerów 600 i 1200), interfejsem SCSI (zewnętrzne dyski dla komputerów A1500, A2000, A2500, A3000 i A4000), wbudowany na stałe jak w Amidze CDTV lub w specjalnie zamiawianych konfiguracjach komputerów o oznaczeniach wyższych niż 2000.

Dysk twardy - odznacza się pojemnością wielokrotnie wyższą niż dla dyskietki (80MB-4GB) oraz znacznie szybszym działaniem. Znakomicie przyspiesza pracę lub wręcz ją umożliwia. Większość nowych programów jest wydawana jedynie w wersji instalacyjnej zakładając, że używanie dyskietek jako podstawowego nośnika mija się z celem. Dysk twardy jest automatycznie rozpoznawany jedynie w systemach 1.3 i nowszych. Zależy to również od kontrolera dysku, gdyż nie wszystkie automatycznie "montują" dysk po włączeniu zasilania. Warto pamiętać, że zainstalowanie HD zabiera część pamięci operacyjnej na bufory dyskowe. Stąd warto go instalować przy pamięci przekraczającej 1MB, która jest absolutnym minimum. System 2.0 znacznie ułatwia gospodarowanie HD, umożliwia stosowanie FFS dla wszystkich partycji.

Złącza PCMCIA - to uznany na całym świecie sposób podłączania praktycznie dowolnych urządzeń do różnych komputerów. Szczególnie znane są z możliwości podłączania "dysków krzemowych" czyli bateryjnie podtrzymywanych pamięci o pojemnościach 32KB - 4MB. Umożliwiają one bardzo szybkie przenoszenie danych pomiędzy komputerami. Do złącza PCMCIA można podłączyć inne urządzenia jak dysk twardy, stację dysków elastycznych, modemy, karty sieci komputerowych itp. (patrz Tabela 1)

Pamięć i jej rozszerzenia.

Tutaj trzeba rozróżnić podział na trzy typy pamięci. Każda Amiga standardowo posiada pamięć Chip RAM. Amiga 500 jako jedyna posiada możliwość zainstalowania pamięci Slow, pozostałe Amigi mają możliwość dokupienia jedynie pamięci Fast oraz Chip (Tabela 2).

W tabelce dwa razy pojawiła się pamięć Chip, po raz pierwszy jako standardowo montowana pamięć, później jako maksymalna ilość jaką można zainstalować. Wróćmy do typów pamięci jakie można upchać w nasz komputerek. Pamięć Chip jest wykorzystywana zarówno do przechowywania programów jak i specjalnych danych dla układów specjalizowanych. Przechowuje więc wyświetlany obraz, szybkie animacje, tworzone bezpośrednio przez procesory graficzne oraz wszelkie dźwięki i muzyczki. Mimo tak wielu zalet jest ograniczona do 1 MB w Amidze 500 i CDTV. W nowych komputerach jej pojemność wynosi maksymalnie 2MB co jest ciągle kroplą w morzu potrzeb. Tak więc kupujemy pamięć Fast. Rozszerzymy ogólną pamięć komputera oraz ... przyspieszymy nasz komputer! W starych Amigach przyspieszenie wynosi około 30-50% , w Amidze 1200 jest nawet trzykrotne. Wynika to z faktu, że pamięć Fast jest używana jedynie przez procesor, [redacted] koprocesory muzyczno-graficzne nie muszą pośredniczyć w dostępie do tego typu rozszerzenia. Pamięć PCMCIA to pamięć Fast w postaci małych modułów (tzw. "karty kredytowe" lub "krzemowe dyskietki") podłączanych do portu PCMCIA.

Z pokazanej tabelki wynika, że najlepsze możliwości rozszerzania ma A1200. Jest to komputer 32-bitowy i podłączenie 16-bitowej karty PCMCIA nie tylko nie przyspieszy jego działania, ale znacznie je spowolni. Stąd też maksymalna "sensowna" konfiguracja pamięci to 6MB. Zobacz również fragment o procesorach.

Grafika

Koputery Amiga mają obecnie trzy typy procesorów graficznych. System wyświetlający obraz jest znacznie bardziej rozwinięty niz w typowych komputerach, składa się z układów wyświetlających obraz, dwóch procesorów graficznych i systemu wyświetlania tzw. "duszków" czyli prostokątnych obiektów graficznych wyświetlanych niezależnie od podstawowego obrazu. Do tego dochodzi unikalna możliwość łączenia różnych trybów graficznych [redacted] jednym ekranie. Jak już wcześniej wspomniałem istnieją trzy generacje układów graficznych:

a). Standardowe
b). Rozszerzone ECS (ang. Enchanced Graphic System)
c). Zaawansowane AGA (ang. Advanced Graphic Architecture)

Tabela 3 pokazuje standardowo instalowane układy oraz możliwość ewentualnej zmiany. W Tabeli 4 pokazuję możliwości poszczególnych układów. W celu lepszej orientacji pokrótce wyjaśnię uży-

Tabela 3

Komputer	Ukł. std	Ukł. roz
CDTV	STD	ECS*
A500	STD	ECS
A500+	ECS	/
A600	ECS	/
A1200	AGA	/

* - Mogą wystąpić poważne trudności z oprogramowaniem na dyskach CD-ROM



te w Tabeli 4 rubryki. Jako maksymalną rozdzielczość podaję tzw. Max overscan czyli maksymalną rozdzielczość, którą bez problemu obsługuje komputer. Można korzystać z jeszcze wyższych, lecz ekran Workbencha nie może być w niej otworzony. Maksymalne wymiary rastra to wirtualny rozmiar ekranu jaki bez problemu mogą obsłużyć koprocesory graficzne. Takie rozmiary może mieć ekran Workbencha (tylko jeśli współpracujesz z systemem 2.0). Na monitorze będziesz widział jedynie wycinek całego ekranu, reszta będzie przesuwana automatycznie zgodnie z ruchem myszki. Dual playfield to specyficzny tylko dla Amigi tryb wyświetlania dwóch różnych obrazów na jednym ekranie. Jeden z obrazów "zakrywa" fragmenty drugiego, będąc przezroczystym w miejscach zdefiniowanych jako jeden z kolorów. W odpowiedniej rubryce zamieściłem ilość kolorów jaką może wykorzystywać każdy z obrazów w tym trybie. Wreszcie ostatnia rubryka pokazuje względną prędkość układów w porównaniu do standardowych układów. W tabelce nie uwzględniłem specjalnego trybu graficznego, który współpracuje z monitorem A2024 przy rozdzielczości 1024x1008 w czterech kolorach. Wprawdzie kości standardowe mogą z nim współpracować, lecz wymagają drobnych przeróbek (zależy to od modelu Amigi i wersji kości w obrębie jednej generacji). Kości ECS I AGA współpracują z A2024 bez problemu, jedynym wymaganiem jest system nowszy niż 1.4 (1.3 + dodatkowe oprogramowanie dołączone na dysku).

Tabela 4

Układy	Standardowe	ECS	AGA
Rozdzielczość	736 x 580 [2]	1450 x 580 [3]	1450 x 580, 640 x 1024
Paleta kolorów	4096	4096	16777216 (16,7 mln)
Max raster	1224 x 1024	32768 x 32768	262140 (HAM8), 16.7mln*
Max il. kolorów	4096 (HAM) [1]	4096 (HAM) [1]	262140(HAM8), 16,7 mln*
Szer. Sprite'ów	16 punktów	16 punktów	32punkty
Kolory Dualplay	8 [1]	8 [1]	16
Względna prędkość	1x	2x	8x

* - korzystając z Coppera

[1] - przy rozdzielczości < 340x512

[2] - maksymalnie 16 kolorów

[3] - maksymalnie 4 kolory z palety 64

Procesor

Każda Amiga wyposażona jest w zestaw spejalizowanych procesorów muzyczno - graficznych oraz jeden procesor główny. Jest to zawsze procesor Motoroli z rodziny MC 680x0. Każdy z procesorów posiada 32 bitową architekturę, 32 bitowe instrukcje oraz 16-to lub 32-dwu bitową szynę danych, którą porozumiewa się ze światem zewnętrznym. Procesory Motoroli jako jedne z nielicznych są w 100% kompatybilne (procesor nowszy bez problemu wykonuje programy przewidziane dla jego poprzedników). Dla porównania - procesory Intel stosowane w komputerach IBM PC są kompatybilne tylko w niektórych aspektach, procesor nowszy wykonując program poprzednika nie może korzystać z rozszerzonych możliwości. Stąd bardzo często szybkie procesory 32-dwu bitowe 80386/486 liczą na danych 16-to bitowych.

Komputery omawiane w tym artykule wykorzystują jedynie dwa typy procesorów Motoroli.

A). MC68000

Procesor ten zapoczątkował całą rodzinę. Instrukcje tego procesora oraz rejestry pozwalają na bezpośrednie korzystanie z danych 32-bitowych. Jedynym ograniczeniem są instrukcje dzielenia i mnożenia. Od pełnej architektury 32-bitowej odróżnia go 16-bitowa szyna danych. Aby przesłać daną 32-bitową musi podzielić ją na dwie 16-bitowe i przesłać "na raty". Operacje na danych 32-bitowych są wykonywane prawie 2 razy wolniej niż te operujące na słowie (16-bitach).

B). MC68010

Procesor ten nie jest stosowany w żadnym z komputerów firmy Commodore. Jest on jednak kompatybilny elektronicznie z poprzednikiem. Wystarczy wyjąć poprzedni układzik i wstawić 68010. Uzyskujemy prawie półtorakrotne przyspieszenie komputera, a w niektórych przypadkach nawet dwukrotne (pętle DBcc, rozkazy mnożenia i dzielenia). Istnieje drobna niekompatybilność z procesorem MC 68000 - instrukcja MOVE from SR jest uprzywilejowana, stąd czasami (bardzo rzadkie przypadki programów, które nie powinny jej używać) występują "przykre niespodzianki".

C). MC68020

To w pełni 32 bitowa konstrukcja. Dzięki "szerszej" szynie danych oraz optymalizacji niektórych instrukcji (między innymi instrukcje przesuwów i rotacji) procesor ten jest ponad dwukrotnie wydajniejszy od poprzedników. Do sprawności komputera przyczynia się pamięć podręczna (ang. Cache), która odciąża szynę dla procesorów graficznych. Niestety jego wydajność znacznie zmniejszają nowe układy AGA. Aby w pełni wykorzystać jego możliwości należy kupić pamięć Fast RAM. Procesor ten posiada jeszcze jedną zaletę - można bez żadnych problemów podłączać koprocesory matematyczne MC68881 lub nowszy MC68882. Mówię koprocesory, gdyż może ich być nawet siedem! (Tabela 5)

Tabela 5

Komputer	Procesor	Rozszerzany	Wzgl. predkość	Wzgl. prędkość z Fast	Zegar
CDTV	MC68000	MC 68010	1	1.30	7,14 Mhz
A500	MC68000	MC 68010	1	1.30	7,14 Mhz
A500+	MC68000	MC 68010	1.20	1.50	7,14 Mhz
A600	MC68000	żaden	1.20	1.50	7,14 Mhz
A1200	MC 680EC20	dowolny	2.5 - 3.0	5-7	14.28 Mhz



Rubryka "Rozszerzany" w Tabeli 5 pokazuje typ procesora na jaki można najprościej (i najtaniej) zamienić standardowo montowany układ. Amiga 1200 pozwala na taką zamianę przez specjalnie przygotowany slot [redacted] dnie komputera. Procesor wstawia się razem z prostą płytką wraz z dodatkową pamięcią Fast oraz np. koprocesorem matematycznym. W przypadku zmiany procesora możemy dodać więcej pamięci Fast niż [redacted] to pozwala standardowa konfiguracja (max 8MB Fast, [redacted] czego 4 MB to wolna pamięć PCMCIA), teoretycznie możliwa jest pamięć 4GB, w praktyce stosuje się do 32-64MB (oferta firmy GVP).



Względna prędkość to porównanie prędkości względem Amigi 500 bez pamięci Fast RAM. "Względna prędkość z Fast RAM" to prędkość po dodaniu tej pamięci.

Szczęśliwie omówiłem już cechy naszych komputerków. Pozostaje wybór jednego spośród nich. Myślę, że dokonasz go sam prównując ich możliwości. Poniżej podaję przykłady "właściwych" zakupów.

a). Kupujący ma bardzo niewiele pieniędzy i nie ma sprecyzowanych zainteresowań. Najtańszym komputerem rodziny Amiga jest bez wątpienia uży- [redacted] A500. Będziesz ją mógł kupić nawet o 1/3 taniej niż kolejne modele. Pamiętaj o kupnie modulatora!

b). Możesz sobie pozwolić na nieco więcej, chcesz korzystać z poważniejszych programów. Kup Amigę 600 ewentualnie A500+. Mimo dziwacznej klawiatury sześćsetka posiada kontroler dysku twardego, który stawia ją wyżej niż A500+. Najlepiej od [redacted] znajdź A600 [redacted] 2MB pamięci (to naprawdę minimum!) i HD większym niż 60 MB. Nie opłaca się kupować dysków mniejszych, są niewiele tańsze a przy tym wolniejsze i zawodniejsze.

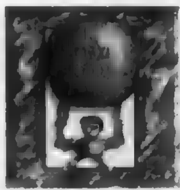
c). Jesteś zwolennikiem gier, czasami używając prostszych programów. Jeśli możesz wysupłać kwotę nieco wyższą niż na A500, doradzam kupno Amigi CDTV. Zwróć przy tym uwagę, aby kupić komplet: CDTV + myszka + klawiatura + pilot. Dyski CD-ROM zapewniają wspaniałą zabawę [redacted] grami i programami edukacyjnymi (encyklopedie, słowniki itd.) jednak musisz mieć dostęp do sklepu który je sprowadza. Jedynym brakiem modelu CDTV jest stary system operacyjny (1.33) i kłopoty [redacted] dodaniem dowolnych rozszerzeń - jeszcze nie są popularne w Polsce.



d). Półprofesjonalnie lub profesjonalnie wykorzystujesz komputery. Jeśli nie stać Cię na wspaniałą A4000 lub np. używaną 3000 (jednak nie do grafiki) to polecam Ci A1200. Jest szybsza od starszych modeli zapewniając zupełnie nową jakość grafiki. Komputer tej klasy wymaga twardego dysku (nie mniej niż 120MB), optymalną konfiguracją jest A1200 [redacted] HD 120MB + 2-4 MB Fast RAM. Nie kupuj Amigi 1200 z 1MB pamięci CHIP (bardzo rzadko spotykane)! Jeśli zajmujesz się grafiką lub inną dziedziną wymagającą dużej mocy obliczeniowej, kup A1200 z wbudowanym koprocesorem matematycznym lub akcelerator o konfiguracji: Fast RAM 2-4MB + MC 68882 25-50 Mhz (+ procesor MC68030 50 Mhz).

e). Uważasz, że możliwości A1200 są nie wystarczające do Twojej pracy. Pozostają Ci dwa rozwiązania - rozszerzyć A1200 do pożądanej konfiguraji np. [redacted] MB Fast RAM + MC [redacted] 50 Mhz + MC 68030 50Mhz + HD 240 MB, lub kupić A4000. Zbytnie rozszerzanie A1200 mija się [redacted] celem, gdyż można przekroczyć cenę A4000 która ma procesor MC 68030/40/60 33/40/50/66Mhz, dwa sloty video, sloty IBM, sloty Zorro III, dysk twardy, koprocesor zintegrowany z procesorem, wygodniejszą klawiaturę, 6 MB pamięci, ...

Życzę udanych zakupów. ❑



Brilliance część 1

Tomasz Łoboda

Historia lubi się powtarzać. To stare, ale niezmiennie aktualne stwierdzenie. Podobnie jak co roku nadejście zimy zaskakuje naszych drogowców, tak i firmy produkujące oprogramowanie dały się zaskoczyć nowym komputerom powstałym w firmie Commodore. Chodzi tu oczywiście o modele A1200, A4000/030 i A4000/040. Przez bardzo długi okres [redacted] praktycznie nie istniał na rynku żaden profesjonalny program potrafiący uporać się z nowymi możliwościami wspomnianych Amig.

Teraz jednak era wymuszonego królowania "Deluxe Painta 4.5" na polu kości AGA dobiegła końca w związku z pojawieniem się nowego programu graficznego o wdzięcznej nazwie: "Brilliance". Czy jednak na pewno?

Aktualna sytuacja przypomina nieco tą, którą znamy już z lat wcześniejszych, kiedy to graficzny świat dzielił się na dwie nacje: zatwardziałych zwolenników serii "Deluxe Paintów" (nie uznających niczego innego) i "eksperymentatorów". Ta druga nacja, do której i ja się zaliczam, była wiecznie niezadowolona z tego co oferowali programiści. Objawiało się to częstymi przeskokami z jednego programu na drugi. A to jeden miał lepiej rozwiązaną paletę barw, a to lepiej wykonywał cieniowanie, [redacted] to szybciej pracował w trybie HAM-6, itd. Wszyscy "eksperymentatorzy" powinni być wreszcie usatysfakcjonowani za sprawą programu "Brilliance".

Filozofia obsługi tego programu nawiązuje do profesjonalnego oprogramowania stosowanego przy kartach graficznych. Wszyscy, którzy pracowali wcześniej z "Digi Paintem" lub "Spectra Colorem" nie powinni mieć kłopotów z szybkim opanowaniem rozbudowanych menu omawianego programu. A jest nad czym popracować (przykład tylko dwóch włączonych opcji został pokazany [redacted] rysunku poniżej). Trzeba jednak przyznać, że ilość i rodzaj oferowanych opcji, w przypadku "Deluxe Painta" (bo to chyba jedyny ogólnie dostępny program, do którego można się odwoływać przy jakichkolwiek porównaniach) i programu "Brilliance", jest prawie jednakowa. Natomiast porównanie prędkości działania przypomina nieco bajkę o wyścigu żółwia ("DPaint") z zającem ("Brilliance"). Oboje dotrą do mety, ale w jakim tempie! Żółw to "siła spokoju", natomiast zając reprezentuje często nonszalancką szybkość. Porównanie to świetnie przystaje do obu programów.

Tyle tytułem wstępu, pora przejść do bardziej konkretnych spraw. "Brilliance" w oryginale to dwa nieco różniące się od siebie programy. Pierwszy [redacted] nich obsługuje zwykłe, znane nam już tryby graficzne, poszerzone o dwie istotne własności: pracę w trybie 256-u kolorów (występujących jednocześnie na ekranie) oraz możliwość ustawienia maksymalnej palety barw [redacted] 16.7 mln kolorów. Drugi natomiast nosi nazwę "TrueBrilliance" i pozwala na rysownie w dwóch specyficznych dla Amigi trybach: HAM-6 (teoretycznie 4096 kolorów) [redacted] HAM-8 (teoretycznie 262144 kolorów na ekranie). Trzeba zaznaczyć, że "Brilliance" jest pierwszym programem graficznym na Amigę, który korzysta w pełni z nowych kości graficznych.

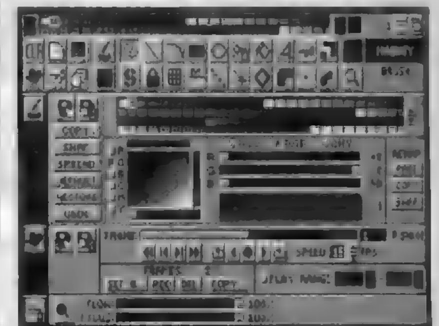
Program zajmuje trzy dyskietki. Na pierwszej znajduje się "Brilliance" wraz z programem instalacyjnym, na drugiej jego wersja "True", natomiast na trzeciej dodatkowe elementy typu: palety kolorów, przykładowe grafiki, brushe, itp. Przy uruchamianiu programu na A1200 bez rozszerzenia pamięci, należy się liczyć [redacted] poważnymi kłopotami związanymi z pracą w wyższych rozdzielczościach. Objawia się to wyłączeniem opcji "Undo" oraz często pojawiającym się komunikatem "Insufficient memory" skutecznie uniemożliwiającym normalną pracę. Na podstawie własnych doświadczeń mogę stwierdzić, że wszystkie opcje są w pełni



Buffer panel

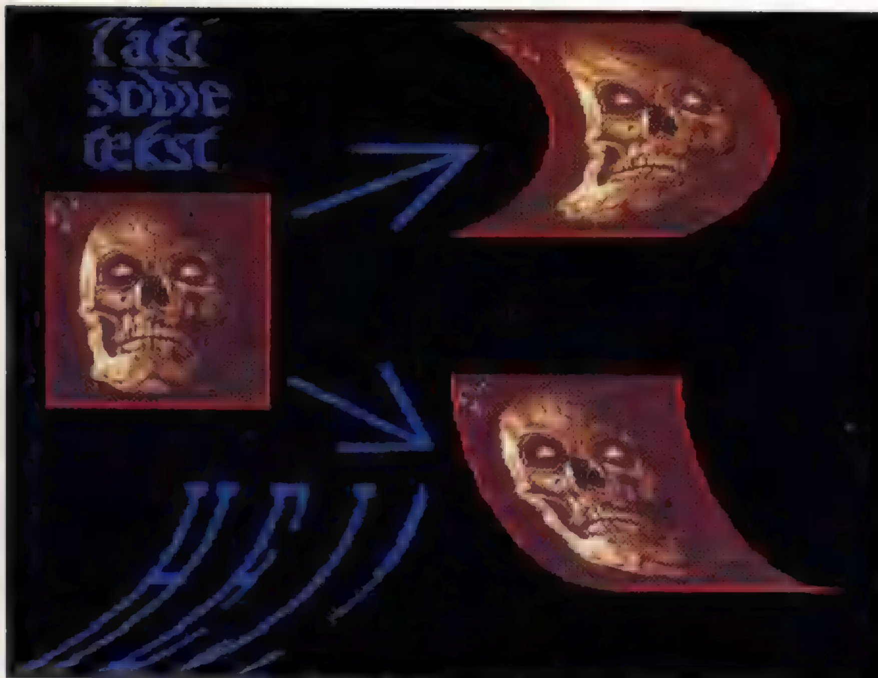


Screen menu



W pierwszym momencie mnogość opcji programu przytłacza użytkownika

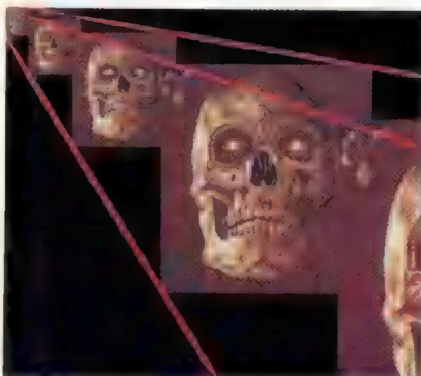




Możliwość opcji: "Bend brush on X"

dostępne jedynie do rozdzielczości maksymalnej wynoszącej 320 na 480 punktów (dla trybu "Multiscan"). Jeżeli chcemy osiągnąć coś więcej, wówczas musimy uciec się do znanego z A500 systemu "kruczków" (niektóre z nich postaram się przedstawić po zakończeniu opisu programu).

Po wczytaniu "Brilliance" wita nas eleganckim panelem, pozwalającym na bezpośredni dostęp do wszystkich opcji programu. Postaram się je przybliżyć potencjalnym użytkownikom.



Przykład działania opcji "Brush size"

Zacznijmy od lewej strony:

{1} "Clear picture" - tej opcji chyba nie trzeba bliżej nikomu przedstawiać, ale dla zasady przypomnę, że powoduje ona usunięcie z ekranu całego rysunku.

{2} "Buffer menu" - podstawowe okno na świat programu. Pozwala na:

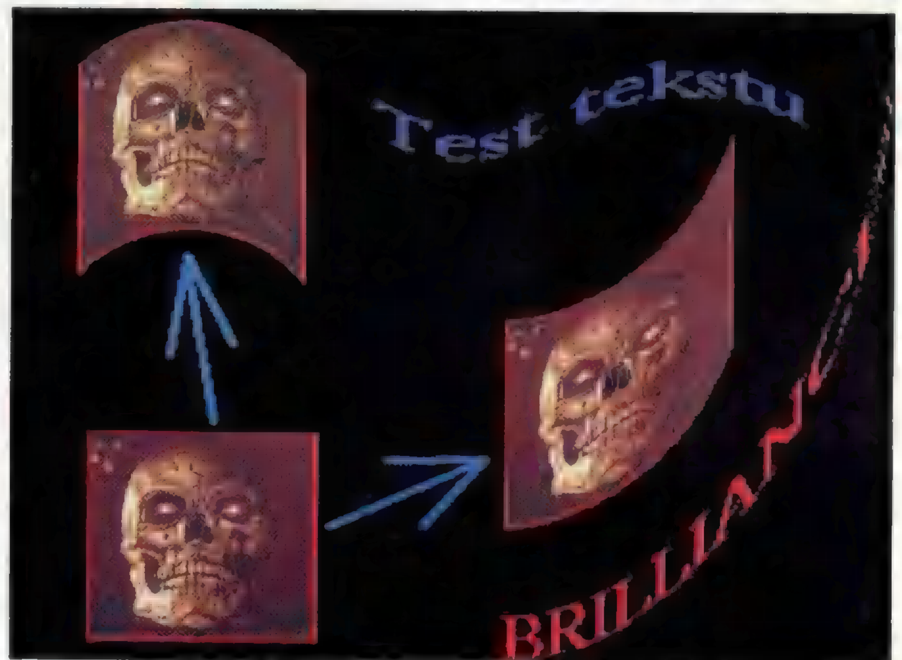
[a] "Load picture" - wczytanie wcześniej stworzonego obrazka,

[b] "Save picture" - zapisanie aktualnie edytowanego bufora,

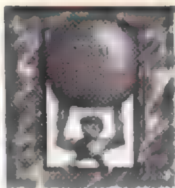
[c] "Preferences" - definiujemy tu niektóre elementy związane z pracą programu. "Autoscroll screens" powoduje automatyczne przesuwanie obrazka w przypadku, gdy dotrzemy kursorem do krawędzi widocznej aktualnie jego części. Kiedy już znajdziemy się na końcu arkusza, wówczas pojawi się specyficzna siateczka symbolizująca kres naszej przestrzeni roboczej. Przypomina to nieco przesuwanie kartki papieru po blacie biurka. Uaktywnienie opcji "Create icons" powoduje automatyczne dodawanie przez program odpowiednich ikonek nagrywanym obrazkom, brushom i animacjom. Wybranie "Relative coordinates" umożliwia chwilowe przeniesienie punktu zerowego koordynat X, Y do wskazanego, pierwszego przy rysowaniu danej krzywej, punktu. Pozwala to między innymi na dokładne określenie wielkości kół, elips, okręgów, prostokątów, itp. Jeżeli nie wybierzemy tej opcji wówczas koordynaty będą mierzone od standardowego początku układu współrzędnych (jest nim lewy górny róg obrazka). "Workbench" uaktywnia system graficzny Amigi. Znajdujący się po prawej stronie "Undo buffer" rezerwuje pewną ilość pamięci dla opcji "Undo" (czyli wycofania się z ostatniej czynności dokonanej w programie). Określenie wielkości tego bufora (opcją "Size") jest równoznaczne z ustaleniem ilości kroków, które mogą być odtworzone wstecz.

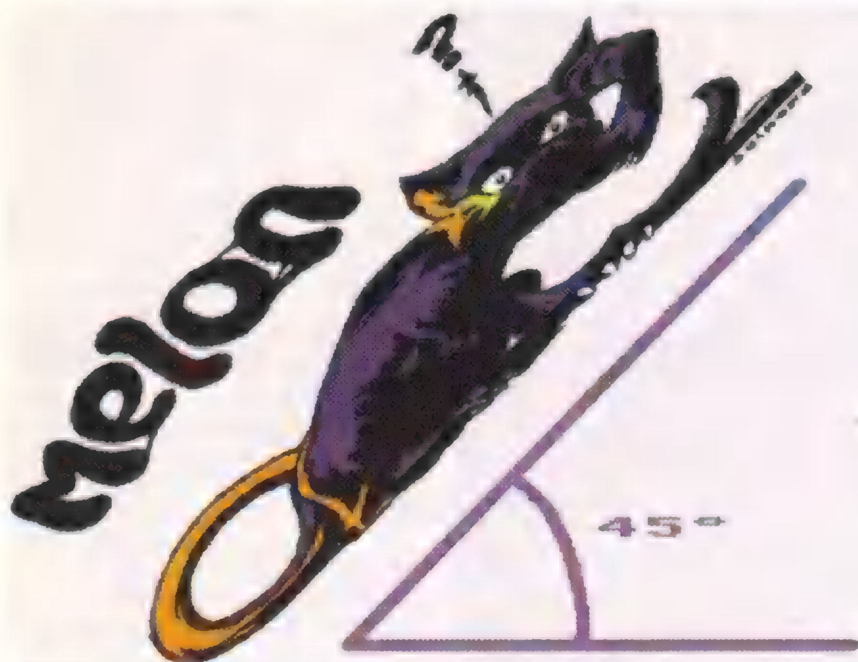
[d] "Print page control" - menu związane z drukiem naszego rysunku. Ze względu na swoją końcową rolę w procesie twórczym zostanie ono omówione na zakończenie opisu,

[e] "Buffer selector" - suwak przełączający aktualnie utworzone bufory



"Bend brush on Y" w akcji





Przykład działania opcji "Brush rotate"

z rysunkami. Program posługuje się dwoma ich rodzajami: "Work" - roboczy i "Spare" - dodatkowy.

[f] "Add buffer" - dodaje nowy bufor (ich ilość jest uzależniona od ilości wolnej pamięci),

[g] "Delete buffor" - kasuje aktualnie wybrany bufor,

[h] "Show page" - pokazuje rysunek w całej okazałości (z aktualnie wybranego bufora),

[i] "Mark" - zmienia status bufora z "Work" na "Spare",

[j] "Go to" - pozwala na natychmiastowy przeskok z bufora roboczego do bufora dodatkowego,

[k] "Copy" - kopiuje zawartość bufora roboczego do bufora dodatkowego i odwrotnie.

[l] "Front" i "Back" - pozwalają kopiować do aktualnego bufora zawartość bufora znajdującego się bezpośrednio przed nim ("Front") lub bezpośrednio za nim ("Back").

Opcje "Quit" i "About" to odpowiednio: wyjście z programu oraz informacje o jego autorach.

{3} "Screen settings" - tutaj możemy określić podstawowe parametry ekranu roboczego, czyli: "Screen mode" - tryb pracy monitora, "Screen size" - wielkość ekranu, "Page size" - wielkość uwidocznionej części rysunku. Suwak "Colors" ustala maksymalną ilość kolorów równocześnie widocznych na ekranie, natomiast "Out of" definiuje wielkość dostępnej palety barw.

Przedstawione powyżej opcje pozwalają nam przygotować program do właściwej pracy, czyli rysowania. Aby uprościć zapis wprowadźmy pewne uproszczenia, a mianowicie: 1*lpm będzie oznaczało pojedyncze naciśnięcie lewego przycisku myszy, natomiast 1*ppm określa tę samą operację w odniesieniu do prawego przycisku myszy. Jest to o tyle ważne, że większość opcji w programie spełnia inne funkcje przy wskazywaniu lpm niż przy naciśnięciu ppm. Można również przypisać obu klawiszom różne kolory z palety. Wskazuje się je na palecie barw {4}, z którą powiązane są: [a] - ("Expand palette") wywołuje całą aktualną paletę barw oraz [b] - ("Scroll palette") przesuwa schowaną paletę o jeden wiersz do góry lub w dół. O aktualnie wybranych kolorach informuje okienko [c], którego górna połowa odpowiada lpm, natomiast dolna ppm.

Powróćmy do opcji rysunkowych:

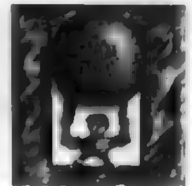
{5} 1*lpm ("Draw") - kreśli przy ciągłym ruchu myszki linię przerywaną. 2*lpm ("Connected draw") - kreśli przy ciągłym ruchu myszki linię ciągłą. 3*lpm ("Filled shape") - rysowana krzywa zostaje zamknięta i wypełniona aktualnym kolorem (istnieją również inne formy wypełniania, które ze względu na swój ogólniejszy charakter zostaną przedstawione przy opisie opcji związanych z brushami i cieniowaniem).

{6} 1*lpm ("Line") - kreślimy prostą przez wskazanie jej punktu początkowego i końcowego. 2*lpm ("Connected lines") - jest to linia prosta kontynuowana. Kreślimy ją wskazując kolejne punkty załamań. 3*lpm ("Filled lines") - działa analogicznie jak "Connected lines", przy czym powstała krzywa zostaje zamknięta i wypełniona kolorem.

{7} 1*lpm ("Elliptical curve") - krzywe eliptyczne. Wskazujemy, trzymając wciśnięty lpm, punkt początkowy i końcowy krzywej, a nastęnie wyginamy ją zgodnie z naszym życzeniem i przycis



Ekslibris powstały poprzez wycięcie kształtem brusha fragmentu rysunku, w aktualnie wybranym kolorze (na rys. był to akurat kolor czarny)



kamy lpm. 2*lpm ("Bezier curve") - niezastąpione w chwili obecnej krzywe Beziera. Definiujemy lpm punkty krańcowe krzywej oraz dwa charakterystyczne punkty krzywizny. Możemy je dowolnie przemieszczać, a ich ostateczne położenie zatwierdzamy ppm.

{8} 1*lpm ("Rectangle") - kreślenie prostokątnej ramki. 2*lpm ("Filled rectangle") rysuje płaszczyznę. Jeżeli naciśniemy ppm wówczas pojawi się "Rectangle menu" określające sposób rysowania prostokąta: za pomocą połowy przekątnej ("Center to corner") lub za pomocą całej przekątnej ("Corner to corner").

{9} 1*lpm ("Ellipse") - kreślimy elipsę. 2*lpm ("Filled ellipse") wówczas rysowana elipsa będzie wypełniona kolorem. 1*ppm powoduje przejście do "Ellipse menu" i pozwala zdefiniować sposób kreślenia elipsy: promieniem ("Center to corner" lub średnicą "Corner to corner". Jeżeli włączymy opcję "Rotate" wówczas po narysowaniu elipsy i przyciśnięciu lpm możemy obrócić ją dookoła środka symetrii.

{10} 1*lpm ("Fine spray airbrush") - "normalny" spray. 2*lpm ("Splatter") - napylamy na rysunek spray składający się z szeregu brushy. 3*lpm ("Shape airbrush") spray będzie napylany w kształcie wyciętego brasha. 1*ppm określi: "Size airbrush"-zasięg sprayu, "Flow"-jego gęstość oraz "Focus"-skupienie wiązki sprayu.

{11} 1*lpm ("Fill") - wypełnia określony obszar wybranym kolorem.

{12} 1*lpm ("Text requester") - określamy tu rodzaj fontu, jego wielkość oraz sposób przedstawienia: "Plain" - normalny, "Bold" - pogrubiony, "Italic" - pochyły i "Underline"- podkreślony.

{13} 1*lpm ("Cut brush") - wycinanie brusha za pomocą prostokątnej ramki. 2*lpm ("Carve brush") wycinanie brusha dowolną krzywą.

{14} 1*lpm ("Brush distort") - pozwala dokonywać prostych operacji na brushu, takich jak: skalowanie ("Size"), poprzez jego dwukrotne zwiększenie względem X,Y ("Double X,Y"), dwukrotne zmniejszenie brusha względem X,Y ("Halve X,Y") lub dowolną zmianę wielkości ("Adjust"). Opcja "Flip X,Y" oferuje odpowiednio: utworzenie symetrycznej figury do aktualnego brusha względem osi X oraz analogicznie względem osi Y. "Bend X,Y" to "wyciąganie" środka brusha w kierunku zgodnym z osią X lub osią Y. "Shear X,Y" pozwala "pochylić" brusha poprzez przesunięcie jednego z jego boków lub płaszczyzn, w kierunku zgodnym z osią X lub osią Y. "Rotate" powoduje obrót brusha o określony (lub dowolny w przypadku "Adjust") kąt, w płaszczyźnie rysunku. "Fast90" to obrót o 90 połączony ze zmianą proporcji. "Outline" dodaje, a "Trim" usuwa ramkę wokół wyciętego brusha. Opcja "Restore" przywraca jego stan wyjściowy.

{15} 1*lpm ("Undo") - cofnięcie ostatniej operacji. 1*ppm ("Redo") jest to opcja przeciwna do "Undo".

{16} 1*lpm ("Magnify") - powiększa wybrany obszar ekranu.

[1] - wyłączenie powiększenia,

[2] - zwiększenie wartości powiększenia,

[3] - zmniejszenie wartości powiększenia,

[4] - włączenie/wyłączenie gradacji powiększonego obszaru,

[5] - wycofanie się z ostatniego działania.

{17} 1*lpm ("Pen tip") - aby cokolwiek narysować na ekranie, opcja ta musi być uaktywniona (wyjątek stanowi rysowanie wyciętym wcześniej brushem). Po wskazaniu ppm pojawia się "Pen tip menu", w którym wybieramy podstawowy kształt pędzla. Znajdują się tam także dwa przyciski ("Size pen tip") pozwalające dowolnie przeskalować zarówno pędzle okrągłe, jak i prostokątne.

{18} 1*lpm ("Anti alias") - włącza specjalną technikę mieszania barw, dzięki której oko ludzkie przestaje zauważać tzw. schodkowość rysowanych na ekranie krzywych (co jest związane z wielkością pojedynczych pikseli monitora). 2*ppm powoduje przejście do "Anti alias menu", a w nim mamy możliwość określenia dokładności pracy opcji "Anti alias" (od "Minimum", czyli minimalnej przez "Low" - niską, "Medium" - średnią po "High" - najwyższą).

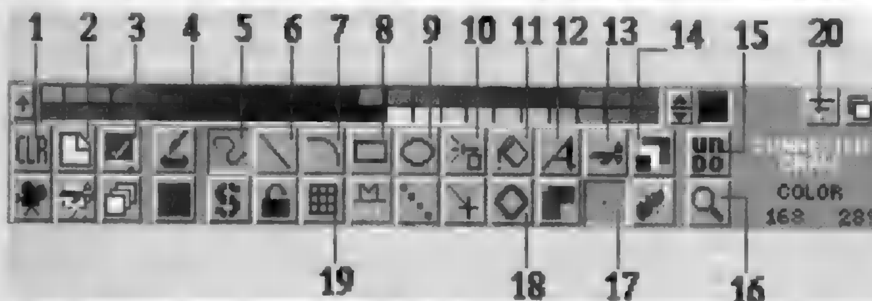
{19} 1*lpm ("Grid menu") - powoduje przeskoki kursora graficznego po niewidocznej siatce, której wielkość ustawiamy za pomocą "Grid width" i "Grid height" po wykonaniu 1*ppm na przycisku {19}. "Grid X offset" i "Grid Y offset" określają przesunięcie siatki względem punktu zerowego układu współrzędnych. "Adjust" umożliwia ustawienie wielkości siatki bezpośrednio na ekranie, za pomocą myszki. Bardzo ciekawe możliwości daje opcja "Brush", ustalająca przeskoki kursora zgodnie z wielkością aktualnie używanego brusha.

{20} ("Squash menu") - 1*lpm chowa dodatkowy, otwarty aktualnie panel (warto wspomnieć, że definitywne zamknięcie takiego panela następuje po 1*lpm lub 1*ppm na pierwszym przycisku z jego lewej strony). Powtórne wskazanie {20} przywraca poprzedni stam rzeczy.

Na tym kończę pierwsze nasze spotkanie z "Brilliancem". Do tej pory starałem się omówić jedynie podstawowe opcje rysunkowe programu. Pozostałe elementy (takie jak: animacja, dodatkowe operacje na brushach, praca z paletą barw, wypełnienia, itp.), ze względu na ich skomplikowanie, postaram się przedstawić w drugiej części opisu w postaci krótkich przykładów.

PS.

Podawanie w nawiasach angielskich nazw poszczególnych funkcji nie jest bynajmniej popisywaniem się autora wątpliwą znajomością tego języka, lecz ma znacznie głębszy cel. Otóż twórcy programu postanowili ułatwić życie początkującym użytkownikom i wygospodarowali prawą część panelu głównego informacje o aktualnie wybranej lub gotowej do wybrania opcji. Pozwala to na łatwe opanowanie programu oraz daje możliwość zapoznania się z pewną specyficzną terminologią, która może się później przydać podczas pracy z innymi programami graficznymi. W części informacyjnej panelu znajdują się również informacje o sposobie ry sowania i koordynaty kursora graficznego. ❑



Panel główny programu



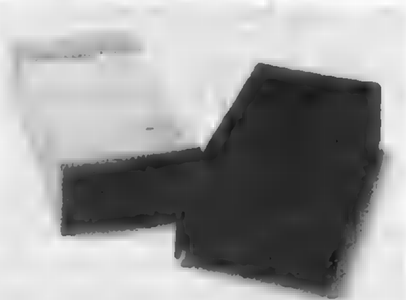
Amigowe ABC część 2

Tomasz [illegible]

Jeśli myślisz, że Twój komputer jest sam w sobie wystarczająco niesamowity i właściwie nie istnieje już nic więcej, co mogłoby Cię zaszokować, to szczerze mówiąc, niewiele jeszcze wiesz o Amidze. Z kilkoma ulepszeniami sprzętowymi możesz przekształcić ten komputer w coś zupełnie innego. Prawdziwy kombajn, który nie tylko odsłoni przed Tobą zupełnie nowe możliwości tego wspaniałego komputera, ale również pozwoli wykorzystać go w wielu nowych dziedzinach. Teraz postaram się opisać kilka podstawowych urządzeń peryferyjnych i protez do Amigi...

Zacznijmy od podstaw:

Dodatkowa stacja dysków



Jeżeli zapytasz przeciętnego użytkownika Amigi, jakie urządzenie peryferyjne dokupił do komputera jako pierwsze, najczęściej odpowie on, że była to dodatkowa stacja dysków. Zakup taki nie jest zresztą czystym luksusem, lecz przy pracy z Shell'em (że nie wspomnę o kopiowaniu dyskietek) staje się raczej absolutną koniecznością. Wiele aktualnie produkowanych programów zajmuje więcej niż jedną dyskietkę. Wykorzystanie dodatkowej stacji dysków znacznie ułatwia ich użycie, ponieważ zwalnia nas z przykrego obowiązku ciągłego zmieniania dyskietek. Do Amigi, dzięki zastosowaniu przelotek, można podłączyć trzy dodatkowe stacje dysków.

Rozszerzenie pamięci

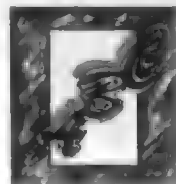


Pamiętam, kiedy kilka lat temu zamieniłem swojego małego Commodore C-64 na Amigę, wielkość 0.5 MB pamięci tkwiącej we wnętrzu tego komputera, wydawała mi się wprost niewyobrażalna. Myślałem wtedy (wspominając marne 64Kb mojego poprzedniego komputera), że właściwie nic nie może jej zapełnić. Jednak już po kilku dniach mogłem przekonać się jak bardzo się myliłem. Obecnie komputery wyposarzone w 1Mb umożliwiają uruchomienie jedynie gier i kilku prostrzych aplikacji. Jeżeli myślisz o poważniejszym wykorzystaniu Amigi, np. do prac graficznych, video czy DTP, musisz liczyć się z dodatkowym wydatkiem, właśnie na rozszerzenie pamięci. Jego wielkość zależy oczywiście od Twoich potrzeb i możliwości finansowych, ale 4Mb powinieneś uznać za podstawę - absolutne minimum niezbędne do poważnej pracy.

Dysk twardy



Dysk twardy jest kolejnym logicznym krokiem po dokonaniu zakupu dodatkowej stacji dysków. Jego użycie znacznie zwiększa komfort korzystania z komputera. Duża pojemność i szybkość dostępu do danych sprawiają, że praca z komputerem nabiera nowego wymiaru. Każdy, kto kiedykolwiek używał twardego dysku, nie zechce z niego zrezygnować! Do Amigi dostępne są twarde dyski o pojemnościach: 20, 30, 40, 50, 60, 80, 100, 120, 180, 240MB, itd. Od razu chciałbym przestrzec przed nabywaniem tych o najmniejszej pojemności. Faktycznie wartość 20 czy 30MB może wydawać się jak na początek wystarczająca, jednak już po kilku tygodniach (ba, nawet dniach, czy godzinach) użytkowania może okazać się, że potrzeba znacznie więcej. Wiele programów łącznie z danymi zajmuje po kilkadziesiąt dyskietek (np. Imagine, PageStream, ProPage, itp.). Osobiście radziłbym zakup dysku o pojemności wię



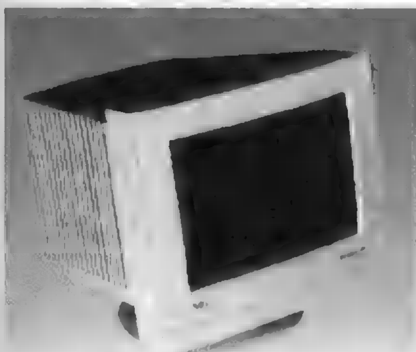
kszej jak 100 MB, choć przy "oszczędnym" wykorzystaniu wystarczy i 40-50Mb.

Drukarka



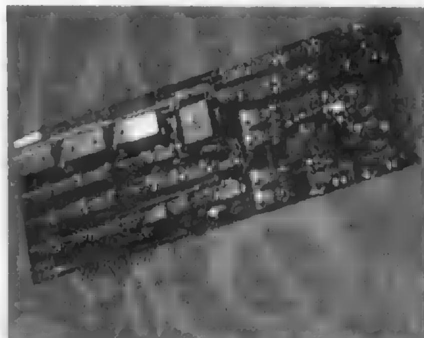
Jeżeli pragniesz zatrzymać "namacalny" dowód swojej pracy na komputerze, drukarka jest dla Ciebie koniecznością. Jest to urządzenie, które przy odpowiednim wykorzystaniu może przydać się każdemu. Jeżeli jesteś programistą, uzyskasz wydruki programów, jeżeli pracujesz z tekstem - wydruki tekstów, jeżeli muzykujesz - wydruki nutowe. Jeżeli natomiast bawisz się grafiką czy DTP - drukarka (i [redacted] nie byle jaka) jest Ci absolutnie niezbędna. Rozróżniamy trzy podstawowe typy drukarek: igłowe (9 lub 24), atramentowe i laserowe.

Monitor



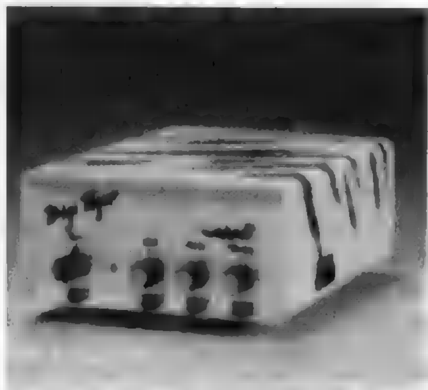
Jeżeli wykorzystujesz Amigę wyłącznie w celach zabawowych, monitor nie jest Ci właściwie potrzebny - wystarczy komputer podłączony za pomocą modukatora TV do zwykłego telewizora. Jeżeli jednak myślisz o czymś poważniejszym, powinieneś zastanowić się nad zakupem monitora RGB. Dzięki podłączeniu uzyskać można obraz wysokiej jakości, nieporównywalny z obrazem uzyskiwanym na ekranie telewizora. Użycie monitora znacznie ułatwia wykonywanie wszelkich operacji (zwłaszcza graficznych) i podwyższa komfort pracy.

Karta Turbo



Domowe modele Amigi są stworzeniami, mimo ich wielu zalet, potwornie powolnymi, co przy profesjonalnym użyciu staje się ogromnie uciążliwe. Można jednak przyspieszyć działanie komputera stosując specjalne karty turbo. Dzięki ich zastosowaniu można sprawić, że zwykła "pięćsetka" będzie pędziła z szybkością porównywalną do A4000. Zanim jednak dokona się zakupu takiej karty, trzeba zastanowić się, czy nie lepiej dokonać zamiany samego komputera. Może się okazać, że [redacted] karty turbo jest mniejsza od szybszego modelu Amigi.

Digitizery obrazu



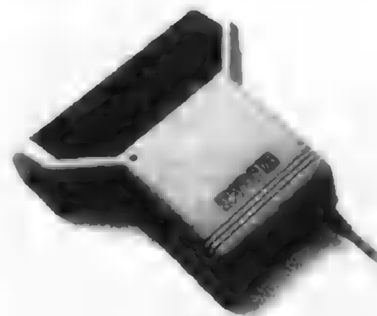
Są to urządzenia pozwalające na przeprowadzenie konwersję analogicznego sygnału wizyjnego na dane cyfrowe odczytywalne przez komputer. Dzięki ich zastosowaniu możemy "wczytać" do komputera i uzyskać na ekranie monitora obraz dostarczony [redacted] podłączonej kamery lub magnetowidu. Rozróżniamy dwa rodzaje digitizerów: szybkie i wolne. Pierwsze [redacted] nich, dzięki zastosowaniu specjalnego bufora pamięci mogą przeprowadzać operację konwersji obrazu [redacted] bardzo dużą szybkością. Umożliwia to "ściąganie" nawet całych sekwencji filmowych w czasie rzeczywistym. Są to jednak, w przeciwieństwie do digitizerów "wolnych", urządzenia bardzo drogie. Digitizery wolne są prostrze w budowie i, [redacted] tym idzie, znacznie tańsze. Niestety ich możliwości są nieco ograniczone. Umożliwiają one ściąganie jedynie obrazów statycznych z kamery lub stopklatki magnetowidu.

Genlock



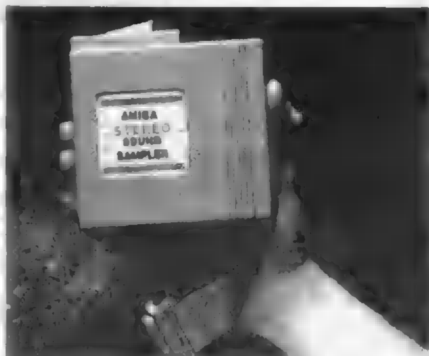
Są to urządzenia służące do nakładania obrazu z komputera na obraz dostarczony z kamery lub innego źródła sygnału wizyjnego. W połączeniu z bardzo bogatym oprogramowaniem oferowanym dla Amigi, mogą zostać wykorzystane zarówno w domu do upiększania własnej filmoteki rodzinnej, jak i w celach nieco bardziej profesjonalnych - w małych studiach telewizji osiedlowej. Zresztą [redacted] tu dużo mówić - wiele stacji telewizji regionalnych w naszym kraju używa Amigi do tych właśnie celów.

Scannery



To urządzenia pozwalające na ściąganie obrazu z wszelkiego rodzaju dokumentów, zdjęć, plakatów, itd. Rozróżniamy scannery ręczne i stacjonarne. Można je jeszcze zróżnicować pod względem kilku innych parametrów, takich jak ilość rozpoznawanych kolorów, czy maksymalna rozdzielczość skanowania. Na ogół scannery stacjonarne posiadają wyższe parametry skanowania niż ręczne, są oczywiście wygodniejsze w użyciu i o wiele droższe. Cena może dochodzić nawet do kilkudziesięciu tysięcy marek przy skanerze dobrej klasy. Ceny skanerów ręcznych nie przekraczają 400 DM.

Samplery



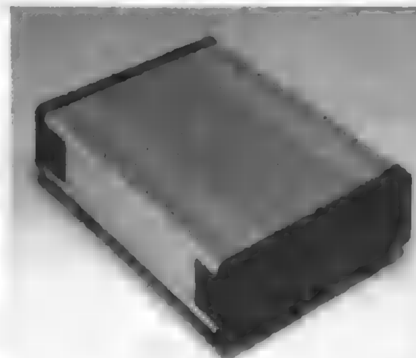
Sampler jest prostym i stosunkowo tanim urządzeniem przetwarzającym sygnał analogowy z wejścia na sygnał cyfrowy dla komputera. Umożliwia on tworzenie własnych efektów dźwiękowych, dem muzycznych, "przegrywanie" muzyki z magnetofonu na komputer. Jest to podstawowe narzędzie każdego muzyka pozwalające eksperymentować z dźwiękiem. Dzięki specjalistycznemu oprogramowaniu możliwe jest poddanie dalszej obróbce wczytanego do pamięci sampla (próbki dźwięku) Dzięki podłączeniu do Amigi samplera można zamienić ją również w cyfrową kamerę pogłosową i źródło wielu ciekawych efektów akustycznych.

MIDI Interface



Jest urządzeniem stanowiącym pomost umożliwiający współpracę elektronicznych instrumentów muzycznych z komputerem. Dzięki odpowiedniemu oprogramowaniu można znacznie rozszerzyć możliwości posiadanego instrumentu, wzbogacić jego brzmienia, wyposażyć go w sequencer.

Modemy



Są urządzeniami służącymi do przyłączenia komputera do sieci telefonicznej w celu połączenia się z innymi użytkownikami komputerów. Rozwiązanie to umożliwia bezpośrednie połączenie między dwoma posiadaczami komputerów, wymianę danych, programów i w sumie świetną zabawę. W Polsce istnieje już cała "sieć" modemowców skupiona wokół tzw. BBS-ów - czyli punktów zbiorczych, gdzie wszyscy podsyłają ciekawe prace i informacje, skąd wszyscy mogą z nich korzystać. □

Turbo System A500/500+

Turbo System został zaprojektowany z myślą o tych użytkownikach popularnego u nas w kraju komputera A500/+, którzy wykorzystują go również do pracy!

TS oparty jest na procesorze MC 68000-12F taktowanym podwójną częstotliwością zegara tzn. 14.28MHz. Oczywiście procesor taktowany znamionowo 2 razy szybciej nie mógłby pracować w systemie przystosowanym do pracy wolniejszej. Dlatego też oprócz szybszego procesora wymagany jest elektroniczny układ synchronizacji pozwalający na poprawną pracę systemu.

Wielkość przyspieszenia uzyskiwanego w czasie pracy komputera (procesora) jest ściśle zależna od aktualnej konfiguracji sprzętowej. Najmniejszą wydajność Turbo System przejawia przy pracy w standardowej Amidze wyposażonej w KickStart 1.3, bez dodatkowej pamięci FAST. Jego wydajność zależy tutaj przede wszystkim od rodzaju programu i wykorzystaniu przez program czasu procesora.

Raptowne poprawienie wydajności Turbo Systemu uzyskujemy przez zainstalowanie dodatkowej pamięci FAST RAM już w podstawowej wielkości 2MB. Dalsze przyspieszenie pracy systemu uzyskujemy przy pracy pod KickStart'em 2.04, który realizuje wówczas szybciej obsługę niektórych procedur systemowych.

Układ został zaprojektowany z uwzględnieniem montażu modułu KickStart na płycie komputera.

Pomiary szybkości wykonane przy pomocy programu SysINFO V3.14 wykazywały przyspieszenie nie mniej niż 200%. Dalszą poprawę prędkości pracy uzyskamy przy zastosowaniu procesora MC68010.

ABC asemblera odcinek 4

Adam Gregorowicz

W dzisiejszym ABC ... poznamy jak zwykle garść nowych rozkazów i nauczymy się, jak uzyskać najprostrzą grafikę przy wykorzystaniu procedur zawartych w systemie operacyjnym Amigi.

Z poprzedniego odcinka winien jestem jeszcze pełen opis instrukcji AND.

AND - AND logical (Iloczyn logiczny.)

Zapis: AND Dn,przeznaczenie lub AND źródło,Dn
Rozakaz AND wykonuje iloczyn logiczny na operandzie źródłowym i przeznaczenia, a wynik umieszczany jest w operandzie przeznaczenia. Dokładny opis iloczynu logicznego znajduje się w trzecim odcinku niniejszego kursu. Rozmiarem instrukcji AND może być bajt, słowo lub długie słowo.
Znaczniki: X - Nie zmieniany.
N - Ustawiany, gdy najstarszy bit wyniku jest równy jeden, w przeciwnym razie zerowany.
Z - Ustawiany, gdy wynikiem operacji jest zero. W przeciw nym wypadku zerowany.
V - Zawsze zerowany.
C - Zawsze zerowany.

Jak wynika z opisu składni rozkazu AND, jeden z operandów (źródłowy lub przeznaczenia) musi być rejestrem danych. Tak więc, gdy operandem źródłowym jest rejestr danych, dozwolone są następujące tryby adresowania:

```
AND Dn,Dm
AND Dn,(Am)
AND Dn,(Am)+
AND Dn,-(Am)
AND Dn,p(Am)
AND Dn,p(Am,Ro.W)
AND Dn,p(Am,Ro.L)
AND Dn,nn.W
AND Dn,nn.L
```

Natomiast, gdy rejestr danych jest operandem przeznaczenia możemy korzystać z takich oto trybów adresowania:

```
AND Dm,Dn
AND (Am),Dn
AND (Am)+,Dn
AND -(Am),Dn
AND p(Am),Dn
AND p(Am,Ro.W),Dn
AND p(Am,Ro.L),Dn
AND nn.W,Dn
AND nn.L,Dn
AND p(PC),Dn
AND p(PC,Rm.W),Dn
AND p(PC,Rm.L),Dn
AND #n,Dn
```

Przykład: AND.B D0,D1 powoduje wykonanie iloczynu logicznego na ośmiu bitach (rozmiar bajtu) rejestrów danych D0 i D1. Wynik zostaje umieszczony w rejestrze D1. Przykładowo, gdyby w rejestrze D0 była wartość $5F (binarnie %01011111), a w D1 wartość $B3 (binarnie %10110011), to wynikiem operacji byłaby wartość $13 (binarnie %00010011). Dokładniej przedstawia to rysunek.

```
     01011111
AND  10110011
-------------
     00010011
```

Oprócz intrukcji AND istnieje jeszcze instrukcja ANDI różniąca się od poprzedniej krótszym czasem wykonywania i możliwością wykonywania iloczynu logicznego na rejestrze statusowym.

ANDI - AND immediate (Natychmiastowy iloczyn logiczny.)

Zapis: ANDI #n,przeznaczenia
Rozkaz ANDI wykonuje operację iloczynu logicznego (dokładnie w ten sam sposób co AND) na liczbie będącej operandem źródłowym i operandzie przeznaczenia. Rozmiarem operacji może być bajt, słowo lub długie słowo. Jak zawsze w przypadku, gdy operandem źródłowym jest dana natychmiastowa, należy pamiętać, że rozmiar tej danej i rozmiar operacji muszą być takie same.

Wspomniałem, że rozkaz ANDI umożliwia modyfikowanie rejestru statusowego. Dostęp do tego rejestru jest możliwy dwoma sposobami:
- Jeśli operacja jest w rozmiarze bajtu, to zmieniany jest je-



dynie dolny bajt rejestru statusowego zawierający znaczniki.

- Gdy operacja ma rozmiar słowa, wtedy możliwa jest zmiana wszystkich szesnastu bitów rejestru statusowego. Jest to jednak operacja uprzywilejowana, co znaczy, że może być wykonana jedynie w trybie nadzorcy.

Znaczniki: X - Nie zmieniany.

N - Ustawiany, gdy najstarszy bit wyniku jest równy jeden, w przeciwnym razie zerowany.

Z - Ustawiany, gdy wynikiem operacji jest zero. W przeciwnym wypadku zerowany.

V - Zawsze zerowany.

C - Zawsze zerowany.

Dozwolone tryby adresowania:

```
ANDI #n,Dn
ANDI #n,(An)
ANDI #n,(An)+
ANDI #n,-(An)
ANDI #n,p(An)
ANDI #n,p(An,Rm.W)
ANDI #n,p(An,Rm.L)
ANDI #n,nn.W
ANDI #n,nn.L
ANDI #n,SR
ANDI #n,CCR
```

Przykład: ANDI.B #1,CCR powoduje wykonanie operacji iloczynu logicznego na bajcie użytkownika rejestru statusowego (zawierającym znaczniki). Ponieważ na zerowym bicie znajduje się znacznik C (Carry), to jego wartość zostanie zachowana, natomiast pozostałe znaczniki zostaną wyzerowane (dokładny opis bajtu użytkownika i rejestru statusowego znajduje się w pierwszym odcinku tego kursu - Amigowiec 3-4/93).

To na tyle, jeśli chodzi o zaległości z poprzedniego odcinka, przejdźmy teraz do rzeczy nowych.

Z mojej praktyki wynika, że samo poznanie działania poszczególnych rozkazów procesora nie wystarcza do pisania programów w języku maszynowym. Konieczne jest również poznanie (i to dosyć dokładne) systemu operacyjnego komputera, na jakim mamy zamiar pisać programy.

Przyjmijmy, że chcemy narysować na ekranie kratę o czterech polach w poziomie i ośmiu polach w pionie. Aby tego dokonać musimy najpierw znaleźć jakiś sposób wyświetlenia czegoś na ekranie. Otóż system operacyjny Amigi zawiera bibliotekę intuition.library umożliwiającą nam skorzystanie z okienka, na którym możemy sobie dowolnie rysować. No tak, ale co to takiego jest biblioteka. Biblioteka to zbiór procedur napisanych w języku maszynowym i odpowiedzialnych za obsługę jednego typu zadań (np. graphics.library zajmuje się grafiką, dos.library - obsługą systemu operacyjnego .Procedury te znajdują się w tak zwanym Kikstart ROMie Amigi lub są w razie potrzeby dogrywane do pamięci z dysku z katalogu libs: (takie biblioteki to np. bardzo popularna powerpacker.library lub mathtrans.library). Najważniejszą z bibliotek jest biblioteka exec, której zadaniem jest kontrola całego systemu, obsługa przerwań, jak i otwieranie oraz zamykanie innych bibliotek, gdyż przed użyciem procedury z jakiejś biblioteki (za wyjątkiem funkcji biblioteki exec) należy ją najpierw otworzyć. My otworzymy najpierw bibliotekę intuition.library, gdyż to w niej zawarte są procedury obsługi okienek.

```
_LVOOpenLibrary  equ -552

OpenIntuition:
    move.l 4.W,a6              ; baza biblioteki exec
    lea IntuitionName(pc),a1   ; nazwa biblioteki intuition
        ; zakończona zerem
    moveq #0,d0                ; nr wersji biblioteki
    jsr _LVOOpenLibrary(a6)    ; otworzenie biblioteki
    move.l d0,IntuiBase        ; zapamiętanie adresu
                               ;bazowego biblioteki intuition
    rts

IntuitionName dc.b 'intuition.base',0,0 IntuiBase dc.l 0
```

Aby otworzyć intuition.library do rejestru a6 wpisaliśmy zawartość długiego słowa spod adresu 4. Jest tam zawarty tak zwany adres bazowy biblioteki exec. Wywołanie jakiejkolwiek funkcji z tej (i innej) biblioteki dokonywne jest instrukcją:

```
jsr przesunięcie(a6)
```

"Przesunięcie" to adres względny (oczywiście względem adresu bazowego), pod jakim znajduje się skok do określonego podprogramu w bibliotece. Każdy z tych adresów jest szczegółowo opisany (np. w "Amiga ROM Kernel Reference") i posiada własną nazwę (zawsze zaczynającą się od _LVO). Tak więc używając procedur systemu nie trzeba pamiętać szeregu liczb, wystarczy operować nazwami (które przecież łatwiej zapamiętać niż liczby). Trzeba jednak posiadać pliki, w których każdej nazwie przypisana jest odpowiednia wartość. Ponieważ jednak w moim kursie stosowanie bibliotek jest nam potrzebne jedynie do zobrazowania bez zbędnego wysiłku wyników działania pracy programów, będę podawał dla każdej procedury odpowiadające jej przesunięcie (np. _LVOOpenLibrary equ -552).

Kolejnym krokiem jest załadowanie do rejestru A1 adresu nazwy biblioteki (Uwaga! Nazwa biblioteki musi być pisana małymi literami!), a do rejestru D0 numeru wersji biblioteki (w tym przypadku nie jest ważna wersja biblioteki, więc wpisujemy zero). Po wpisaniu właściwych wartości do A1 i D0, wywołujemy funkcję OpenLibrary. Procedura ta zwraca w rejestrze D0 adres bazowy otwieranej biblioteki, który najlepiej jest gdzieś zapamiętać. Gdyby w D0 było zero, oznaczałoby to, że żądana biblioteka nie została odnaleziona.

W podobny sposób musimy otworzyć bibliotekę graphics.library (zawiera ona funckje dotyczące rysowania linii, okręgów, itp.) oraz dos.library (wykorzystamy z niej funkcję odczekującą określony czas). Spróbujcie sami napisać właściwe podprogramy otwarcia bibliotek, a potem porównajcie efekt swojej pracy z programem z pierwszego listingu (podprogram OpenGraphics: i początek podprogramu Wait:).

Przy wychodzeniu z programu będziemy musieli zamknąć wszystkie użyte biblioteki (nie jest to absolutnie konieczne, ale na wszelki wypadek lepiej tak zrobić). Oto przykładowy fragment programu zamykający bibliotekę intuition.library:

Gdy wiemy już jak otworzyć biblioteki możemy zabrać się do otwarcia okienka:

```
_LVOCloseLibrary equ      -414

CloseIntuition:
  move.l 4.W,a6              ; baza biblioteki exec
  move.l IntuiBase(pc),a1    ; baza biblioteki intuition
  jsr _LVOCloseLibrary(a6)   ; zamknięcie biblioteki
  rts
```

```
_LVOOpenWindow equ -204

OpenWindow: move.l IntuiBase(pc),a6 ; baza biblioteki intuition
  lea WindowDef(pc),a0          ; adres definicji okna
  jsr _LVOOpenWindow(a6)        ; otworzenie okna
  move.l d0,WindowBase          ; zapamiętanie adresu struktury
       ; Window
  rts
```

Otwieramy je używając procedury OpenWindow z biblioteki intuition.library, której podajemy w A0 adres struktury NewWindow. Musimy ją sami w pełni stworzyć. OpenWindow założy strukturę Window i w D0 zwróci nam jej adres. Gdyby z jakichś powodów coś się nie udało (np. za mało pamięci), to w D0 będzie zero.

W ten sposób otworzyliśmy okna, jednak aby coś w nim narysować potrzebny nam jest adres struktury RastPort. Znajdziemy go w strukturze Window ■ polu wd_RPort:

```
move.l WindowBase(pc),a1   ; adres stuktury Window
move.l wd_RPort(a1),a1     ; adres struktury RastPort
```

Gdy mamay już potrzebne adresy, możemy przejść do rysowania odcinka. Będziemy do tego potrzebować dwóch procedur z biblioteki graphics.library. Dlaczego dwóch? Otóż linia jest zawsze rysowana od aktualnej pozycji kursora graficznego do pozycji podanej jako parametry dla funkcji Draw. Do ustawienia kursora graficznego służy procedura Move, która w A1 wymaga adresu RastPort, ■ w D0 i D1 odpowiednio x i y pozycji kursora. Właściwą procedurą rysującą odcinek jest Draw, która w A1 również wymaga adresu struktury RastPort okna, w którym chcemy rysować, natomiast w D0 i D1 - współrzędnych x i y końca rysowanej kreski.

A oto całkowity podprogram rysujący linię od współrzędnych podanych w D0, D1 od współrzędnych z D2 i D3:

```
_LVOMove equ -240 _LVODraw equ -246
DrawLine:                        ; d0 - x początkowe
       ; d1 - y początkowe
       ; d2 - x końcowe
       ; d3 - y końcowe
  movem.l d0-a6,-(sp)            ; zapamiętanie wszystkich
       ; rejestrów na stosie
  move.l GraphBase(pc),a6        ; baza biblioteki graphics
  move.l WindowBase(pc),a1       ; adres stuktury Window
  move.l wd_RPort(a1),a1         ; adres struktury RastPort
  movem.l d2/d3/a1,-(sp)         ; zapamiętanie na stosie
                                 ; zawartości rejestrów
                                 ;D2 D3 i A1
  jsr _LVOMove(a6)               ; wywołanie funkcji Move
  movem.l (sp)+,d2/d3/a1         ; zdjęcie ze stosu rejestrów
  move.w d2,d0                   ; x końcowe do d0
  move.w d3,d1                   ; y końcowe do d1
  jsr _LVODraw(a6)               ; wywołanie funkcji Draw
  movem.l (sp)+,d0-a6            ; zdjęcie wszystkich
       ; rejestrów ze stosu
  rts
```

Zwróćcie uwagę na zapamiętywanie potrzebnych rejestrów na stosie przed wywołanie funkcji systemu. Nie wiemy przecież, czy procedura systemu nie zmienia zawartości jakiś rejestrów.

I jeszcze bardzo ważna rzecz - zamknięcie okna:

```
_LVOCloseWindow equ -72

CloseWdw:
  move.l IntuiBase(pc),a6       ; baza biblioteki intuition
  move.l WindowBase(pc),a0      ; adres struktury Window
  jsr _LVOCloseWindow(a6)       ; zamknięcie okna
  rts
```

Po tym przydługim, ale koniecznym, wprowadzeniu możemy zająć się właściwą procedurką rysującą kratkę. Na

Opis struktury NewWindow

+00	słowo LeftEdge	pozycja lewej krawędzi okna względem lewej krawędzi ekranu
+02	słowo TopEdge	pozycja górnej krawędzi okna względem górnej krawędzi ekranu
+04	słowo Width	szerokość okna
+06	słowo Height	wysokość okna
+08	bajt DetailPen	kolor detali (np. gadżetów głębokości)
+09	bajt BlockPen	kolor tła
+10	długie słowo IDCMPFlags	znaczniki IDCMP (zbyt skomplikowane, aby coś więcej o tym pisać)
+14	długie słowo Flags	znaczniki określające wygląd okna
+18	długie słowo Gadget	adres struktury gadżetów dołączonych do tego okna
+22	długie słowo Image	adres grafiki używanej zamiast "ptaszka" w rozwijanych menu
+26	długie słowo Title	adres nazwy okna
+30	długie słowo Screen	adres struktury Screen, na którym mają otworzyć się okna, dla okna workbencha - 0
+34	długie słowo BitMap	adres struktury BitMap, jeśli chcemy używać własnych bitmapów (lepiej nie chciejmy)
+38	słowo MinWidth	minimalna szerokość okna
+40	słowo MinHeight	minimalna wysokość okna
+42	słowo MaxWidth	minimalna szerokość okna
+44	słowo MaxHeight	maksymalna wysokość okna
+46	słowo Type	typ okna



początku zdefiniujemy sobie współrzędne lewego górnego rogu kratki: XP equ 40 YP equ 10

Następnie szerokość i wysokość jednego pola w kracie: SZER equ 40 WYS equ 20

Jak zapewne zauważyliście, szerokość jest dwukrotnie większa od wysokości. Jest to spowodowane tym, że okno otwierane na ekranie workbencha przyjmuje jego rozdzielczość, czyli zazwyczaj 640 na 256, a w tej rozdzielczości punkty nie są kwadratowe, lecz prostokątne.

Przy rysowaniu kratki narysujemy najpierw linie pionowe:

```
DrawKratka:
    move.w  #XP,d0          ; początkowe x i y dla pierwszej
    move.w  #YP,d1          ; pionowej linii kratki
    move.w  #XP,d2          ; końcowe x i y dla pierwszej
    move.w  #YP+8*WYS,d3    ; pionowej linii kratki
    move.w  #5-1,d4         ; jest 5 linii pionowych Loop01:
    bsr  DrawLine           ; wywołanie procedury
                            ; rysującej linię
    add.w  #SZER,d0         ; zwiększenie współrzędnej x
        ; początku linii
    add.w  #SZER,d2         ; zwiększenie współrzędnej x
        ; końca linii
    dbf  d4,Loop01          ; zmniejszenie d4 o jeden, jeśli
        ; większe lub równe 0 to skok
```

Jak widać wystarczy najpierw wpisać do rejestrów początkowe współrzędne, a potem:

1. narysować kreskę,
2. zwiększyć współrzędne x końca i początku linii,
3. spraw dzić, czy jeszcze trzeba rysować, i jeśli tak to wrócić do punktu 1.

Podobnie rysujemy linie poziome:

```
move.w   #XP,d0          ; początkowe x i y dla pierwszej
move.w   #YP,d1          ; poziomej linii kratki
move.w   #XP+4*SZER,d2   ; końcowe x i y dla pierwszej
move.w   #YP,d3          ; poziomej linii kratki
move.w   #9-1,d4         ; jest dziewięć linii poziomych
                         ;Loop02:
bsr      DrawLine        ; wywołanie procedury rysującej
         ; linię
add.w    #WYS,d1         ; zwiększenie współrzędnej y
         ; początku linii
add.w    #WYS,d3         ; zwiększenie współrzędnej y
         ; końca linii
dbf      d4,Loop02       ; zmniejszenie d4 o jeden, jeśli
         ; większe lub równe 0 to skok
rts
```

I to wszystko. Kratka narysowana. Możemy jeszcze spróbować inaczej stworzyć pętlę odliczającą rysowane linie. W przykładzie użyto sekwencji:

```
move.w  #5-1,d4 Loop01:   ..
...
dbf  d4,Loop01
```

Równie dobrze można to zrealizować w inny sposób:

```
move.w  #5-1,d4 Loop01:   ...
...
sub.w   #1,d4
```

Lub tak:

```
move.w  #5,d4 Loop01:   ...
...
sub.w   #1,d4
bne     Loop01
```

Wybór należy do Was. Jasne jest, że pierwszy sposób jest najkrótszy, więc i zapewne najszybszy (oszczędzymy na instrukcji sub.w #1,d4). Tego typu pętle są bardzo zbliżone do znanej z Basic'a pętli FOR ... NEXT z krokiem równym -1 (pascalowe FOR ... DOWNTO).

Jeśli już zrozumieliście (albo i nie) teoretycznie to, co właśnie przeczytaliście, wpiszcie sobie program z listingu 1 i zaobserwujcie w praktyce działanie omawianych procedur. Program ten zawiera nie opisywaną szczegółowo procedurę Wait, ale jej funkcją jest jednynie odczekanie pięciu sekund po narysowaniu kratki, abyście mogli ją sobie spokojnie obejrzeć.

```
;           ## Listing 1 ##

;     ## procedury exec.library ##
_LVOOpenLibrary     equ  -552
_LVOCloseLibrary equ     -414

;     ## procedury intuition.library ##
_LVOOpenWindow    equ  -204
_LVOCloseWindow   equ  -72

wd_RPort  equ  50
WINDOWDRAG equ  2       ; okno można przesuwać po ekranie
GIMMEZEROZERO  equ  1024 ; współrzędna (0,0) nie będzie
                         ; w lewym górnym rogu okna, ale
                         ; w lewym górnam rogu jego części
                         ; roboczej (wewnętrznej)
WBENCHSCREEN   equ  1    ; okno na ekranie workbencha

;     ## procedury graphics library ##
_LVOMove      equ  -240
_LVODrawequ  -246

;     ## procedura dos.library ##
_LVODelay     equ  -198

      section   Krata,code       ; poinformowanie assemblera, że
                                 ; teraz będzie program, a nie
                                 ; dane (bez tego też będzie dobrze)

Start:      jmp  Main            ; skok do procedury głównej


;     ## Procedury otwarcia i zamknięcia bibliotek oraz okna ##

OpenGraphics:
      move.l    4.W,a6                 ; baza biblioteki exec
      lea   GraphicsName(pc),a1        ; nazwa biblioteki graphics
      moveq     #0,d0                  ; nr wersji biblioteki
      jsr   _LVOOpenLibrary(a6)        ; otworzenie biblioteki
      move.l    d0,GraphBase           ; zapamiętanie adresu bazowego
                                       ; biblioteki graphics
      rts

OpenIntuition:
      move.l    4.W,a6                 ; baza biblioteki exec
      lea   IntuitionName(pc),a1       ; nazwa biblioteki intuition
      moveq     #0,d0                  ; nr wersji biblioteki
      jsr   _LVOOpenLibrary(a6)        ; otworzenie biblioteki
      move.l    d0,IntuiBase           ; zapamiętanie adresu bazowego
                                       ; biblioteki intuition
      rts

CloseGraphics:
      move.l    4.W,a6                 ; baza biblioteki exec
      move.l    GraphBase(pc).a1       ; baza biblioteki graphics
      jsr   _LVOCloseLibrary(a6)       ; zamknięcie biblioteki
      rts

CloseIntuition:
      move.l    4.W,a6                 ; baza biblioteki exec
```

ASM

```
    move.l   IntuiBase(pc),a1         ; baza biblioteki intuition
    jsr   _LVOCloseLibrary(a6)        ; zamknięcie biblioteki
    rts

OpenWindow
    move.l   IntuiBase(pc),a6         ; baza biblioteki intuition
    lea   WindowDef(pc),a0            ; adres definicji okna
    jsr   _LVOOpenWindow(a6)          ; otworzenie okna
    move.l   d0,WindowBase            ; zapamiętanie adresu struktury Window
    rts

CloseWdw:
    move.l   IntuiBase(pc),a6         ; baza biblioteki intuition
    move.l   WindowBase(pc),a0        ; adres struktury Window
    jsr   _LVOCloseWindow(a6)         ; zamknięcie okna
    rts

WindowDef:   dc.w 200                 ; współrzędna x okna
    dc.w 20                           ; współrzędna y okna
    dc.w 240                          ; szerokość okna
    dc.w 200                          ; wysokość okna
    dc.b 0                            ; kolor detali w oknie
    dc.b 1                            ; kolor wypełnień w oknie
    dc.l 0                            ; znaczniki IDCMP
    dc.l  WINDOWDRAG+GIMMEZEROZERO    ; znaczniki określające
                                      ;wygląd i typ okna
    dc.l 0                            ; adres pierwszej struktury
                                      ; gadget dla tego okna
    dc.l 0
    dc.l  WindowName                  ; adres testu będącego nazwą okna
    dc.l 0                            ; wskaźnik struktury screen dla ekranu,
                                      ;na którym ma się otworzyć okno
                                      ; (w tym przypadku równy zeru,
                                      ;bo używamy ekranu workbencha)
    dc.l 0                            ; adres struktury BitMap, jeśli chcemy
                                      ; używać własnych bitmapów (my nie chcemy)
    dc.w 0                            ; minimalna szerokość okna
    dc.w 0                            ; minimalna wysokość okna
    dc.w 240                          ; maksymalna szerokość okna
    dc.w 200                          ; maksymalna wysokość okna
    dc.w WBENCHSCREEN                 ; typ okna (w naszym przypadku
                                      ; okno na ekranie workbencha)


WindowName:   dc.b  'Okienko z kreskami',0 ; nazwa okna

GraphicsName: dc.b  'graphics.library',0   ; nazwa biblioteki graphics
IntuitionName: dc.b  'intuition.library',0 ; nazwa biblioteki intuition
DosName: dc.b  'dos.library',0             ; nazwa biblioteki dos

GraphBase:    dc.l  0     ; miejsce przechowywania bazy graphics.library
IntuiBase: dc.l  0        ; miejsce przechowywania bazy
                          ; intuition.library
WindowBase:   dc.l  0     ; miejsce przechowywania adresu
                          ; struktury Window dla okienka

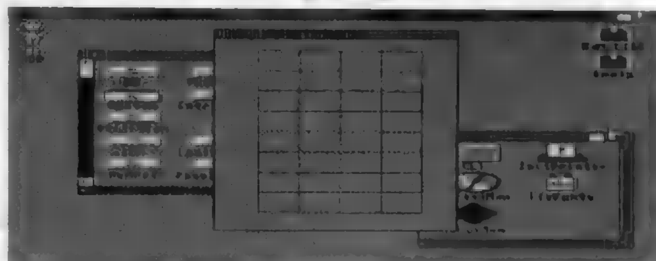
Wait:
    move.l   4.W,a6                   ; baza biblioteki exec
    lea   DosName(pc),a1              ; nazwa biblioteki dos
    moveq   #0,d0                     ; nr wersji biblioteki
    jsr   LVOOpenLibrary(a6)          ; otworzenie biblioteki
    move.l   d0,a6                    ; adres bazowy biblioteki dos do a6
    move.l   #5*50,d1                 ; ilość 1/50 sekundy do odczekania
    jsr   _LVODelay(a6)               ; wywołanie procedury czekającej
    move.l   a6,a1                    ; baza biblioteki dos do a1
    move.l   4.W,a6                   ; baza biblioteki exec
    jsr   _LVOCloseLibrary(a6)        ; zamknięcie biblioteki dos
    rts
DrawLine:          ; d0 - x początkowe
                   ; d1 - y początkowe
                   ; d2 - x końcowe
                   ; d3 - y końcowe
    movem.l   d0-a6,-(sp)             ; zapamiętanie wszystkich rejestrów na
                                      ;stosie
    move.l GraphBase(pc),a6           ; baza biblioteki graphics
    move.l WindowBase(pc),a1          ; adres struktury Window
    move.l wd_RPort(a1),a1            ; adres struktury RastPort
    movem.l d2/d3/a1,-(sp)            ; zapamiętanie na stosie zawartości
                                      ; rejestrów D2 D3 i A1
    jsr   _LVOMove(a6)                ; wywołanie funkcji Move
    movem.l   (sp)+,d2/d3/a1          ; zdjęcie ze stosu rejestrów
    move.w   d2,d0                    ; x końcowe do d0
    move.w   d3,d1                    ; y końcowe do d1
    jsr   _LVODraw(a6)                ; wywołanie funkcji Draw
    movem.l   (sp)+,d0-a6             ; zdjęcie wszystkich rejestrów ze stosu
    rts

;        ## Program główny ##
Main:
    bsr   OpenGraphics     ; wywołanie procedury otwierającej
                           ; bibliotekę graphics
    beq   NoGraphics       ; skok jeśli błąd otwarcia graphics
    bsr   OpenIntuition    ; wywołanie procedury otwierającej
                           ; bibliotekę intuition
    beq   NoIntuition      ; skok jeśli błąd otwarcia intuition
    bsr   OpenWindow       ; wywołanie procedury otwierającej okno
    beq   NoWindow         ; skok gdy błąd otwarcia okna
    bsr   DrawKratka
    bsr   Wait             ; odczekanie pięciu sekund
    bsr   CloseWdw         ; wywołanie procedury
NoWindow:                  ; zamykającej okno
    bsr   CloseIntuition   ; wywołanie procedury zamykającej
NoIntuition:               ; bibliotekę intuition
    bsr   CloseGraphics    ; wywołanie procedury zamykającej
NoGraphics:                ; bibliotekę graphics
    rts                    ; powrót z programu
XP      equ  40            ; zdefiniowanie x początkowego
YP      equ  10            ; zdefiniowanie y początkowego
SZER    equ  40            ; zdefiniowanie szerokości jednego pola kratki
WYS     equ  20            ; zdefiniowanie wysokości jednego pola kratki
DrawKratka:
        move.w   #XP,d0          ; początkowe x i y dla pierwszej
        move.w   #YP,d1          ; pionowej linii kratki
        move.w   #XP,d2          ; końcowe x i y dla pierwszej
        move.w   #YP+8*WYS,d3    ; pionowej linii kratki
        move.w   #5-1,d4         ; jest 5 linii pionowych
Loop01:
    bsr   DrawLine             ; wywołanie procedury rysującej linię
    add.w  #SZER,d0            ; zwiększenie współrzędnej x początku linii
    add.w     #SZER,d2         ; zwiększenie współrzędnej x końca linii
    dbf   d4,Loop01            ; zmniejszenie d4 o jeden, jeśli
                               ; większe lub równe 0 to skok
    move.w   #XP,d0            ; początkowe x i y dla pierwszej
    move.w   #YP,d1            ; poziomej linii kratki
    move.w   #XP+4*SZER,d2     ; końcowe x i y dla pierwszej
    move.w   #YP,d3            ; poziomej linii kratki
    move.w   #9-1,d4           ; jest dziewięć linii poziomych
Loop02:
    bsr   DrawLine             ; wywołanie procedury rysującej linię
    add.w    #WYS,d1           ; zwiększenie współrzędnej y początku linii
    add.w    #WYS,d3           ; zwiększenie współrzędnej y końca linii
    dbf   d4,Loop02            ; zmniejszenie d4 o jeden, jeśli
                               ; większe lub równe 0 to skok
    rts
```

Program ten nagrajcie na dyskietce, gdyż duża jego część będzie wykorzystywana w innych programach przedstawianych w tym kursie (chodzi tu o procedury otwarcia bibliotek i okna). I jeszcze mała informacja. Jeśli używasz assemblera, który pracuje na własnym ekranie (AsmONE, Master Seka), to po zassemblowaniu i uruchomieniu programu nie dziw się, że nie otworzyło się żadne nowe okno. Okienko otworzyło się, tylko, że na ekranie workbencha. Kliknij na gadget zmiany ekranów lub wciśnij lewy Amiga+N i już możesz podziwiać swoją kratkę.

To wszystko w tym odcinku kursu. Następnym razem przedstawię Wam kilka kolejnych instrukcji procesora Motorola 68000 i wykorzystując poznany dziś sposób wyświetlenia czegokolwiek na ekranie, pokażę kilka procedurek, które pomogą zrozumieć ideę programowania w języku maszynowym. ❑

Efekt działania omówionego programu





Pisma, pisemka

Tomasz Łobęda

Czy na Amigę istnieją bazy danych, jeśli tak to jakie? Czy są one kompatybilne z np. dBase III+?

Amiga jest jednym z najwszechstronniejszych mikrokomputerów i oczywiście istnieją (jest ich nawet całkiem sporo) specjalnie przystosowane do niej programy pozwalające na programowanie baz danych. Do najpopularniejszych tytułów należałoby zaliczyć: "AFiler", "ABase", "Flow", "dBMAN", "FoxFile", "XBase" i "SuperBase". Pierwsze trzy tytuły są stosunkowo prostymi programami pozwalającymi na tworzenie niezbyt skomplikowanych baz danych typu: spisy adresów, telefonów, książek, itp. Zadowalają się one minimalną konfiguracją komputera, czyli pracują na Amidze 500 pozbawionej dysku twardego i zaopatrzonej w jedyne 512KB pamięci RAM. Brak dysku twardego powoduje jednak ograniczenie maksymalnej wielkości bazy danych do wielkości odpowiadającej pojemności pojedynczej dyskietki (czyli teoretycznie dla A500 do 880KB). "FoxFile" to amigowski odpowiednik popularnego "FoxBase'a", znanego z PC-etów. "dBMAN" natomiast jest zgodny z kolejnymi wersjami programu "dBase" (konkretnie z wersjami II, III+, IV). Większe możliwości programów "FoxBase" i "dBase" okupione są wyższymi wymaganiami sprzętowymi, tzn. co najmniej 1MB pamięci i dostępem do dysku twardego lub przynajmniej dwóch stacji dysków. Do profesjonalnej pracy najlepiej nadaje się "SuperBase". Najnowszą dostępną wersją tego programu jest "SuperBase Professional v. 4.3". Pozwala on na bezproblemową wymianę plików pomiędzy wszystkimi komputerami na których istnieje "SBase". Jedną z najważniejszych cech "SuperBase'a" jest możliwość korzystania z oddzielnego modułu ("Database management language") pozwalającego na łatwą automatyzację obsługi bazy. Ogólnie można powiedzieć, że jest to swoisty język programowania wzorowany na popularnym Basicu. Dodatkowo istnieje możliwość korzystania z bogatej literatury dotyczącej tej właśnie bazy, gdyż obsługa i programowanie jest na wszystkich rodzajach komputerów jednakowa (oczywiście biorąc poprawkę na pewne specyficzne uwarunkowania sprzętowe).

Osobom nie posiadającym dysków twardych i nie lubiącym nadmiernego żonglowania dyskami, do stworzenia prostej bazy danych, można polecić dowolny edytor tekstów. Programy tego typu posiadają bowiem taką podstawową opcję jak: "Search", czyli wyszukiwanie ciągu znaków z określonego tekstu. Do niezbyt rozbudowanych zastosowań (np. spis programów) w zupełności to wystarcza. Bardziej zainteresowanych tematem, odsyłam do wcześniejszych numerów "Amigowca", w których zagadnienia związane z bazami danych na Amigę pojawiały się dość regularnie, za sprawą artykułów mojego redakcyjnego kolegi Krzysztofa Nowickiego.

Wiadomo, że A500 (ze standardowym wyposażeniem) nie rozpoznaje FFS. Czy znaczy to, że dyskietka zapisana w taki sposób i przeglądana na "File Masterze" nie będzie miała wyodrębnionego katalogu? Jeśli tak, to w jaki sposób da się uzyskać katalog dysku pod systemem 1.3?

Problem związany z zapisywaniem dyskietek w systemie FFS (skrót od: "Fast File System") gościł całkiem niedawno na łamach "Pism, pisemek". Stwierdziłem wówczas, i nadal twierdzę, że niewielkie korzyści płynące ze stosowania FFS-u nie są warte problemów związanych z niestandardowością takiego zapisu. Daje to odpowiedź na pierwszą część pytania. Otóż katalogów zawartych na dysku zapisanym w systemie FFS nie obejrzy się za pomocą programów typu "File Master" (w systemie 1.3). Jest to wykonalne przez zastosowanie specjalnych "nakładek" na system. Jeden z takich programików pojawił się niedawno na wydawanych przez nasze czasopismo dyskach PD i jedyne co mogę doradzić w tym przypadku, to próbę skorzystania z takiego udogodnienia.

Fanatyków oszczędzania miejsca na dyskietkach mogę pocieszyć, że nowsze wersje systemu operacyjnego Amigi (2.0 i 3.0) doskonale radzą sobie z FFSami. Zainteresowanym przypomnę jeszcze, co daje zastosowanie omawianego systemu. Otóż: "powiększa" się pojemność dysku o ok. 40KB oraz, jeżeli już uda się to zrealizować w przypadku systemu 1.3, zostają przyspieszone operacje dyskowe (np. kopiowanie).

Jak w HSPascalu stworzyć własne menu?

Wyczerpująca odpowiedź na to pytanie przekracza zdecydowanie możliwości działu "Pisma, pisemka". Sytuacja powinna nieco rozjaśnić się po uważnym prześledzeniu przykładów znajdujących się na oryginalnej dyskietce z "HSPascalem". Na szczególną uwagę zasługują wówczas znajdujące się na dysku głównym w szufladzie "AmigaDemos" dwie propozycje autorów programu:



"MenuDemo.pas" oraz "MenuUtil.pas". Zapoznanie się z odnalezionymi tam rozwiązaniami powinno pomóc początkującym programistom przy tworzeniu własnych programów.

Czy mając płytę z koprocesorem typu XC [illegible] RC16 (3A79J8604 PGA) i podstawkę pod oscylator mogę założyć oscylator o większej częstotliwości niż 14 MHz? Jaka jest największa częstotliwość i jaki jest do tego potrzebny układ scalony?

Pytanie to odzwierciedla typowo "słowiańską" duszę naszego narodu. Polak nie potrafi żyć bez tzw. "kombinowania", ale to dobrze. Dzięki temu większość znanych naukowców, konstruktorów, czy informatyków pochodzi właśnie z Europy Środkowej. Wracając do pytania. Taka przeróbka jest możliwa, przy czym należy zaznaczyć, że ilość nie udanych operacji tego typu jest zdecydowanie większa od ilości zabiegów zakończonych pełnym sukcesem. O skali trudności związanej z taką przebudową może świadczyć fakt, że wyspecjalizowane firmy serwisowe nie podejmują się tego typu operacji. Aby zachować chociaż minimalne szanse powodzenia, w przypadku podjęcia takiego ryzyka, należałoby powierzyć wspomniane zadanie tzw. "złotej rączce" (w naszym przypadku człowiekowi obdarzonemu wyjątkowymi zdolnościami w dziedzinie elektroniki). Uważam jednak, że ryzyko związane z uszkodzeniem komputera jest zbyt wysokie, aby je podjąć.

W miesięczniku AM [illegible] opisywano w skrócie urządzenie o nazwie "Scan Doubler" firmy CompServ, które załatwia problem monitora do A4000. Czy urządzenie takie jest dostępne również do A1200?

Niestety nie. A1200 podobnie jak dawniej A500 jest modelem o bardzo "zamkniętej" architekturze. Dodatkowo płyta główna wykonana techniką montażu powierzchniowego nie sprzyja jakimkolwiek próbom rozszerzania możliwości komputera. Znając życie można mieć nadzieję, że konstruktorzy pokonają [illegible] problemy. W końcu nie takich "cudów" już dokonywano z naszymi poczciwymi Amigami.

W jaki sposób połączyć Amigi [illegible] sieć komputerową?

Podstawowym warunkiem takiego połączenia jest zaopatrzenie się w odpowiednie karty sieciowe. Następnie należy zakupić właściwe przewody, potrzebne do wzajemnego połączenia komputerów. Ostatnią fazą jest natomiast zainstalowanie oprogramowania pozwalającego na swobodny przepływ informacji pomiędzy poszczególnymi jednostkami sieci. Z pozoru wygląda to prosto, lecz prawidłowe uruchomienie takiej sieci wymaga odpowiedniej wiedzy i doświadczenia, [illegible] także niemałego nakładu finansowego. Korzyści są jednak bardzo duże. Dzięki sieci komputerowej można bezpośrednio przenosić nawet bardzo duże partie danych, a także korzystać z różnych urządzeń zewnętrznych (przez dowolny komputer pracujący w sieci), bez konieczności ich bezpośredniego przyłączania.

W jaki sposób powstaje [illegible] Amidze obraz, chodzi oto, jak często [illegible] wysyłane sygnały synchronizacji, jak długo trwa rysowanie lini itp.?

To kolejny temat rzeka. Amiga od momentu swojego powstania była pomyślana jako komputer do zastosowań graficznych, o czym najlepiej świadczy fakt dobrania częstotliwości pracy procesora w taki sposób, aby był on wielokrotnością częstotliwości pracy sprzętu video. Dzięki zabiegom konstruktorów Amigę można podłączyć, teoretycznie rzecz biorąc, do każdego urządzenia zaopatrzonego w lampę kineskopową. Do najczęściej spotykanych standardów należą oczywiście: stosowany w Europie PAL oraz występujący za oceanem (USA) standard NTSC. Różnią się one ilością wyświetlanych punktów, [illegible] także częstotliwością wyświetlania obrazu (czyli ilością wyświetlanych obrazów na sekundę). I tak odpowiednio dla systemu PAL wyświetlanych jest 50 obrazów na sekundę, natomiast przy systemie NTSC takich obrazków jest 60. Z tymi częstotliwościami związane są dwa podstawowe parametry monitora, potrzebnego do pracy [illegible] odpowiednimi rozdzielczościami. Są to odpowiednio: częstotliwość odchylania poziomego (H - od "Horizontal") i częstotliwość odchylania pionowego (V - od "Vertical") wiązki elektronów, tworzących obraz na ekranie monitora.

Dla odpowiednich systemów wartości te są następujące:

-NTSC- H=15.72 kHz i V=60 Hz,

-PAL- H=15.60 kHz i V=50 Hz,

-EURO36- H=15.76 kHz i V=73 Hz.

Oczywiście wkrótce okazało się, że jest to niewystarczające. Wprowadzono więc do systemu cały szereg dodatkowych "driverów" o następujących przedziałach częstotliwości pracy:

-DblNTSC- H=27.66 do 29.20 kHz i V=58 do 59 Hz,

-DblPAL- H=27.5 do 29.45 kHz i V=48 do 50 Hz,

-EURO72- H=29.32 do 31.43 kHz i V=69 d0 70 Hz,

-Multiscan- H=29.29 do 31.44 kHz i V=58 do 60 Hz.

Czy w monitorze Phillips CM11362 można zwiększyć rozdzielczość? Jeśli tak, to w jaki sposób?

Jak sądzę autorowi pytania chodziło o zlikwidowanie przesławnego interlace'u, czyli nieznośnego migotania obrazu przy rozdzielczościach rzędu 640 na 512 punktów (system PAL) lub 640 na 400 punktów (system NTSC). Z tego typu monitorem, niestety nie da się tego obejść. Są dwie przyczyny takiego stanu rzeczy. Pierwsza [illegible] nich wynika z konstrukcji samego komputera (autor pytania dysponuje Amigą 500). Zastosowano tu bowiem znany z telewizji sposób obsługi ekranu, polegający na wyświetlaniu obrazu z częstotliwością 50 razy na sekundę, przy czym najpierw wyświetlane są tylko linie nieparzyste, a dopiero później linie parzyste. W efekcie tego pojedyncza linia wyświetlana jest tylko 25 razy na sekundę (przez [illegible] odpowiadający 1/50 sekundy) co zostaje odebrane przez nasz wzrok jako migotanie ekranu. Można oczywiście temu zaradzić stosując urządzenie zwane "Flicker Fixerem", które podwaja częstotliwość wyświetlania obrazu. Jednak takiego zabiegu nie wytrzyma telewizor, czy też zwykły monitor. Należy wówczas zaopatrzyć się z monitor standardu VGA lub Multisync. ❑



Amigowy Public Domain #9

Tomasz Glanc

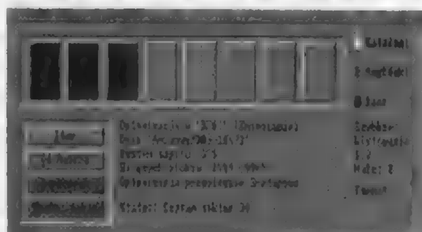
Oto kolejna dyskietka Amigowego Public Domain. Jej zawartość odbiega nieco od dotychczas utrzymywanego przez nas układu, ale [redacted] to programy na niej prezentowane zasługują na uwagę nawet najbardziej wybrednych użytkowników Amigi. A oto dokładny opis zawartości dyskietki.

1. Optymizer dysków. Wersja 3.0b

Jest to kolejna wersja programu znanego już naszym stałym czytelnikom. Ukazał się on na dyskietce PD#8 w 1992 roku. Jednak od tego czasu autor wprowadził do programu wiele rewolucyjnych usprawnień całkowicie zmieniając jego pierwotną postać. Jest to zauważalne zaraz po uruchomieniu programu - główny ekran został całkowicie zmieniony, co znacznie ułatwia obsługę programu i kontrolę całego procesu optymizacji.

Program ten jest wynikiem trwającej już ponad rok ewolucji - poczynając od wersji 1.0a, przez wersję 2.0a (i niewiele się od niej różniące 2.0b i 2.0c) aż do 3.0a i obecnej 3.0b.

Obecna wersja została napisana praktycznie całkowicie od nowa. Algorytm programu został znacznie rozbudowany i przyspieszony. Około pół roku ciężkiej pracy dało rewelacyjny efekt. Program został napisany w 98% w języku C. Pozostałe 2% stanowią wstawki w asemblerze, które znacznie przyspieszają proces optymizacji. Kod źródłowy programu jest zapisany w 7 plikach, o łącznej długości ponad 185 KB (niemal [redacted] tys. linii).



Program ten służy do optymizacji (lub, jak kto woli, optymalizacji) dysków.

Zapis katalogów i plików w systemie AmigaDOS cechuje się tym, że jest bardzo oszczędny pod względem miejsca na dysku. Wadą tego rozwiązania jest niewielka prędkość listowania katalogu i czasami odczytu plików. Optymizer powoduje zmianę ułożenia plików i katalogów na dysku, przez co powolne normalnie procesy zostają znacznie przyspieszone (oszczędza się też mechanikę stacji dysków).

Program potrafi optymizować zarówno dyski zapisane w formacie **OFS**, jak i **FFS**. Radzi sobie też bez problemu z ich odmianami *'International'* oraz *'DirCache'*.

Program ten można uruchomić zarówno z Workbench'a, jak i z CLI (Shell). W tym drugim przypadku program odłączy się (tzw. detach) od okna CLI - znów pojawi się prompt ('1> ') - możesz uruchomić z niego kolejny program lub nawet je zamknąć.

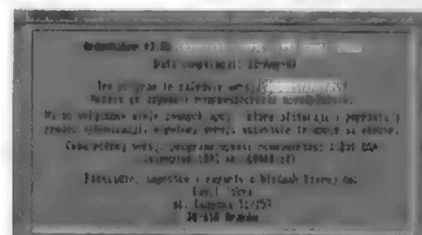
Optymizer spróbuje następnie odczytać plik **'S:OrderMaker.prefs'**. W nim są przechowywane preferencje programu. Jeżeli plik ten nie zostanie znaleziony, to zostaną ustawione preferencje standardowe. Następnie zostanie otwarte okno główne programu. Przed jego otwarciem zostanie przeanalizowane ustawienie kolorów na Workbenchu - dzięki temu bez względu na to, jakiego ustawienia kolorów używasz, gadżety w programie będą miały trójwymiarowy wygląd.

Program można uruchomić na Amigach z systemem w wersji 1.2 lub wyższym.

Aplikacja pracuje prawidłowo już przy Amidze z 0.5 MB RAM, lecz pełnię swych możliwości (szybkość) pokaże dopiero wtedy, gdy dysk źródłowy zostanie cały "na raz" wczytany do pamięci, co przy całkowicie zapełnionej danymi dyskietce wymaga 1.5 MB RAM. Do pracy wystarcza zupełnie jedna stacja dysków, choć oczywiście używanie programu przy większej ilości napędów jest wygodniejsze.

Program był testowany przede wszystkim na mojej Amidze, tzn: A 500, OS 1.3/2.04, 1.5 MB RAM, CPU 68000, 1 FFD 3.5" (Chinon). Był również testowany na innych Amigach, jak np. A1200.

Na dyskietce znajduje się jedynie wersja demonstracyjna programu. Jest ona dość drastycznie "obciętą" wersją programu - nie ma [redacted] niej możliwości włączenia wielu opcji, w ramce wyboru napędu wyświetlane są tylko dwie pozycje itp. Na dłuższą metę uniemożliwia to współpracę z programem, choć uruchamiając tę wersję można ocenić dwie podstawowe cechy programu:





szybkość i skuteczność. Wersja demonstracyjna jest rozpowszechniana na prawach FREEWARE, tzn. można ją używać i rozpowszechniać nieodpłatnie tak długo, jak się komukolwiek podoba.

Zakupu pełnej wersji programu można dokonać bezpośrednio u jego autora (szczegóły w dokumentacji [redacted] dysku).

Autorem programu jest Kamil Iskra.

2. PolFontBootBlock v2.0

Pol Font bootblock wykonuje następujące czynności:

- Uniemożliwia uruchomienie się systemu pod **NTSC**.
- Dopisuje do fontów **TOPAZ/8** polskie znaki pod następującymi kodami:

ą-226 ć-234 ę-235 ł-238 ń-239
ó-243 ś-244 ż-251 ź-250
Ą-194 Ć-202 Ę-203 Ł-206 Ń-207
Ó-211 Ś-212 Ż-219 Ź-218

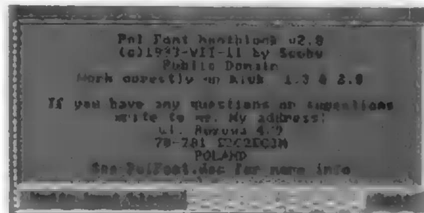
- Zmienia układ klawiatury. Polskie znaki uzyskuje się przez następujące kombinacje klawiszy:

Ą-SHIFT+ALT+a	ą-ALT+a
Ć-SHIFT+ALT+c	ć-ALT+c
Ę-SHIFT+ALT+e	ę-ALT+e
Ł-SHIFT+ALT+l	ł-ALT+l
Ń-SHIFT+ALT+n	ń-ALT+n
Ó-SHIFT+ALT+o	ó-ALT+o
Ś-SHIFT+ALT+s	ś-ALT+s
Ż-SHIFT+ALT+z	ż-ALT+z
Ź-SHIFT+ALT+x	ź-ALT+x

- Jeżeli podczas wczytywania bootblocku zostanie przytrzymany prawy przycisk myszy, zostanie wykonany Hard Reset - cała pamięć zostanie wyczyszczona, ewentualne wirusy znajdujące się w niej zostaną zniszczone (zniszczeniu ulegnie również resetoodporny ramdysk **RAD:**)

PolFont bootblock można zainstalować [redacted] każdym dysku zapisanym w formacie AmigaDOS (nie należy go instalować na dyskach z niektórymi grami). Instalacji można dokonać [redacted] pomocą programu PolFontBootInstal lub przy pomocy Virus Experta (należy najpierw dołączyć PolFont BootBlock do biblioteki bootblocków).

PolFont bootblock został przetestowany pod OS 1.3 i 2.0. Powinien również poprawnie działać na innych wersjach systemu. Polskie znaki tworzone są bez względu na zastosowane w danym systemie fonty.



Pol Font boot block jest programem Public Domain. PolFontBootInit należy kopiować wraz ze zbiorem PolFont.doc.

Autorem programu jest Rafał Mantiuk.

3. QuickDic v1.6



QuickDic jest słownikiem polsko-angielskim i angielsko-polskim. Rozpoznaje ok. 9500 słów polskich i tyleż [redacted] słów angielskich. Pracuje w multitaskingu zarówno w systemie 1.3 jak i 2.0. Przystosowany jest do pracy, nawet w tak ekstremalnych warunkach jak jedna stacja dysków, brak twardego dysku oraz 512KB pamięci. Dane dla słownika zajmują zaledwie 200KB. Program, w stu procentach, został napisany w asemblerze.

Słownik ten umożliwia, w prosty sposób, przetłumaczenie dowolnego słowa, dopisnie i skasowanie nowego wyrazu. Szybkość tłumaczenia uzależniona jest tylko od posiadanego nośnika pamięci, [redacted] w przypadku użycia opcji "Load Dic" (wczytanie danych dla słownika do pamięci), tłumaczenie jest natychmiastowe. Ponadto program posiada takie udogodnienia jak:

- rozpoznawanie przyrostków angielskich,
- syntezator mowy,
- formy odmiany czasowników nieregularnych,
- możliwość wyprowadzania przetłumaczonych wyrazów na dowolne urządzenie np. drukarkę, stację dysków, konsole.

Jeżeli nie posiadasz tablicy kodów do programu to musisz zadowolić się wersją demonstracyjną (rozpoznaje jedynie słowa zaczyniające się na literę A lub B, uniemożliwia dopisywanie wyrazów), lub nabyć taką tablicę. Aby ją otrzymać wystarczy wysłać równowartość 3$ na adres:

ul. Różowa 4/9
70-781 SZCZECIN

Wraz [redacted] powyższą kwotą należy załączyć dokładny adres.

Program można wywołać zarówno z **CLI** jak i z Workbencha. Z **CLI** program wywołuje się wpisując: *QuickDic [<path>]*, gdzie "path" jest nazwą ścieżki wskazującej do katalogu, w którym powinny znajdować się następujące zbiory:

- *Dic.QDIC* - zawiera dane dla słownika
- *Irregular_verbs* - tablica czasowników nieregularnych
- *[Dic.bak]* - poprzednia wersja danych (patrz zapis słownika)

Przy uruchamianiu [redacted] Workbencha w polu "*Tool Types*" ikony można wpisać: *CD=<path>*, gdzie "path" [redacted] znaczenie jak powyżej.

Przy pominięciu tych argumentów przyjmowany jest aktualny katalog. Standartowo powyższe zbiory znajdowały się [redacted] podkatalogu **QDIC/**, lecz ze względu na specyfikę Amigowego PD zostało to zmienione.



Po lewej stronie ekranu QuicDica znajdują się dwa okna podpisane "*English*" i "*Polish*". Chcąc przetłumaczyć wyraz angielski, należy kliknąć [redacted] okno "*English*" (lub nacisnąć lewy *Shift*) i odwrotnie (prawy *Shift*). Wpisywany wyraz może zawierać wszystkie litery dopuszczalne dla danego języka (wyraz polski nie może zawierać liter *Q, X* i *V*, zaś angielski *Ć, Ę, Ł, Ń, Ó, Ś, Ż* i *Ź*), oraz znak spacji (odstęp), myślnik (-) i operatory *?*, oraz ***. *?* - zastępuje każdą literę, zaś * dowolną ilość dowolnych liter lub brak znaku np.:



B?G - może oznaczać zarówno wyraz BAG jak i BIG czy BEG,

*P** - oznacza każdy wyraz [redacted] literę P,

*P*A* - oznacza każdy wyraz na P i zakończony na A.

Znaki te nie mogą występować na początku słowa. W przypadku operatorów * i ? tłumaczony jest pierwszy znaleziony wyraz.

Polskie znaki uzyskuje się przez kombinacje następujących klawiszy:

Alt+a - Ą Alt+c - Ć Alt+e - Ę Alt+l - Ł
Alt+n - Ń Alt+o - Ó Alt+s - Ś Alt+z - Ż
Alt+x - Ź

Po załadowaniu słownika do pamięci przez załączenie gadżetu "*Load Dic*", tłumaczenie wyrazów będzie natychmiastowe. Operacje dopisywania i kasowania słów możliwe są do przeprowadzenia tylko gdy opcja "Load Dic jest aktywna. Z drugiej strony załadowanie słownika wymaga ok. 200 kb wolnej pamięci.

Dopisywanie słów

Wyraz należy wpisać w ten sam sposób jak przy tłumaczeniu słów. W przeciwnym oknie pojawi się tłumaczenie wyrazu lub napis "*Not found*" informujący, że słownik nie zawiera tego wyrazu. Po naciśnięciu gadżetu "*New Word*", można dopisać nowe znaczenie do istniejącego wyrazu lub utworzyć nowy.

Kasowanie słów

Wyraz do skasowania należy wpisać tak jak powyżej, a następnie kliknąć na gadżecie "*Delete word*" i potwierdzić swój wybór. Kasowany jest wyraz główny i wszystkie jego znaczenia, o ile [redacted] nie mają innych znaczeń.

Zapis słownika na dysk

Po dokonaniu wszelkich zmian w słowniku (kasowanie i dopisywanie słów) należy go nagrać na dysk. Pole "*Backup Dic*" w menu "*Project*" decyduje czy ma być utworzony plik "*Dic.bak*" zawierający kopię poprzedniego słownika. Sposób ten gwarantuje większe bezpieczeństwo danych, lecz za to wymaga dwa razy więcej miejsca na dysku.

Opcja "*Restore Dic*" służy do odtwarzania poprzedniego słownika ze zbioru "*Dic.bak*".

Status

Po naciśnięciu gadżetu "*Status*", w oknach "*English*" i "*Polish*" pojawi się odpowiednia ilość słów angielskich i polskich znajdujących się w słowniku. Jeżeli jakieś słowa były uprzednio kasowane, to ilość ta może nie być dokładna. Dzieje się tak dlatego, iż przy kasowaniu słów wyrazy nie są usuwane ze słownika, a jedynie pozostawiane są puste miejsca, które przy najbliższej sposobności są zapełniane nowymi słowami. Jeżeli po dopisaniu nowego słowa ilość wyrazów w słowniku nie zmieni się, to najprawdopodobniej wyraz już figuruje w słowniku lub zostało zapełnione miejsce po skasowanym wyrazie.

Rozpoznawanie przyrostków

Jeżeli gadżet "*Suffix*" jest załączony, to rozpoznawane będą w słowach angielskich przyrostki: -d, -ed, -s, -es, -ing, -ly, -ful itp. i w razie konieczności zostaną usunięte. W przypadku wyrazów polskich, przy czasownikach, będzie tworzona strona bierna, o ile wyraz jej wymaga.

Czasowniki nieregularne

Przy załączonym gadżecie "Irregular verbs", przed każdą próbą przetłumaczenia wyrazu, przeszukiwana jest tabela czasowników nieregularnych. Gdy słowo znajduje się w tej tablicy wyświetlane są wszystkie trzy (ewentualnie dwie) formy tego czasownika, a tłumaczona jest forma podstawowa.

Tablica czasowników nieregularnych znajduje się w pliku "*Irregular_verbs*". Jest to plik tekstowy, więc można go edytować w dowolnym edytorze. Należy jednak przestrzegać pewnych zasad:

- słowa muszą być wpisywane DUŻYMI literami,
- poszczególne formy muszą być oddzielone myślnikami ("-"),
- nowa odmiana powinna znajdować się w nowym wierszu,
- gdy czasownik nie posiada jakiejś formy, należy zostawić znak spacji (odstęp).

Syntezator mowy

Syntezator mowy włącza się przez załączenie gadżetu "*Speak*" (w podobny sposób można go wyłączyć). Wyrazy będą automatycznie wymawiane po każdym przetłumaczeniu. Wymowę wyrazu można powtórzyć przez kliknięcie na gadżecie "*Repeat*".

Do poprawnego działania syntezatora niezbędne są pliki: "*Translator.library*" w podkatalogu "*libs/*" i "*narrator.device*" w podkatalogu "*devs/*" (na dysku PD brakuje tego pliku).

Kierowanie wyjścia na urządzenie

Dzięki tej opcji, przetłumaczone słowa można drukować, zapisywać do pliku tekstowego, wyświetlać w oknie konsoli itp. Nazwę urządzenia na które mają być wysyłane słowa, należy wpisać w oknie pod gadżetem "*Irregular verbs*" np.:

PRT: - oznacza drukarkę,

DF0:filename - oznacza zbiór tekstowy, do którego będą zapisywane wyrazy,

CON:x1/y1/x2/y2/name - oznacza konsolę w położeniu x1,y1 na ekranie Workbencha, o wymiarach x2 na y2 i nazwie "name".

Po wpisaniu nazwy, urządzenie należy otworzyć przez załączenie gadżetu "*Open*". Zamknąć urządzenie można przez wyłączenie tegoż gadżetu.

Do wysyłania wyrazów, na urządzenie, służą gadżety "*M-Send*" i "*A-Send*". "*M-Send*" służy do "ręcznego" wysyłania słów przez kliknięcie na tym gadżecie lub użycie klawisza "*Enter*" (na klawiaturze numerycznej), zaś przy załączonym gadżecie "*A-Send*" słowa będą wysyłane automatycznie po każdym przetłumaczeniu.

Do ustalania formy, w jakiej wyrazy wraz z ich tłumaczeniami mają być wysyłane, służy menu "*Output*":

Pol Fonts - jeżeli nie jest załączone, to polskie znaki będą zastępowane przez podobne litery z alfabetu angielskiego np.: zamiast *ó - o*, zamiast *ł - l*. Opcja ta jest przydatna [redacted] przykład, gdy drukarka nie posiada polskich znaków,

Irr.Verbs - jeżeli jest załączone, to oprócz formy podstawowej czasownika nieregularnego, zostaną przesłane jego nieregularne formy,

W-UpperCase - wyraz tłumaczony pisany dużymi literami,

T-UpperCase - tłumaczenie wyrazu pisane dużymi literami,

Uwaga: Przy wysyłaniu do pliku należy pamiętać o zamknięciu zbioru (przez wyłączenie gadżetu "**Open**"*) po zakończeniu tej operacji!*

Menu "Project"

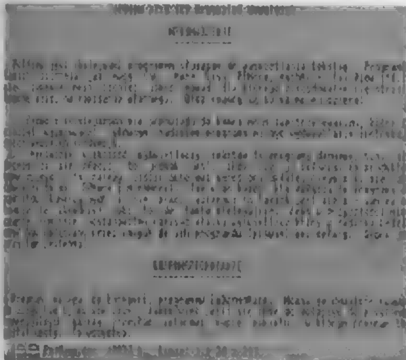
W menu "*Project*", oprócz omówio



nych już pozycji "*Backup Dic*" i "*Restore Dic*", znajdują się "*Sleep*" i "*Quit*". "*Sleep*" powoduje zamknięcie ekranu QuickDica i pojawienie się małego okienka na ekranie Worbencha. Daje to niewielki zysk pamięci. Zaś "*Quit*" powoduje wyjście z programu. Urządzenie, na które były wyprowadzane słowa, zostanie automatycznie zamknięte. Gdy poczyniliśmy jakieś zmiany w słowniku i ich nie nagraliśmy, pojawi się okienko z pytaniem, czy jesteśmy pewni swojej decyzji.

Autorem programu jest Rafał Mantiuk.

4. PolView v1.0



PolView jest (kolejnym) programem służącym do wyświetlania tekstów.

Program należy do kategorii programów ZaDarmoWare. Można go dowolnie używać do czego tam tylko się chce. Jakkolwiek, jeśli się chce go dołączyć do programów komercjalnych należy przesłać autorowi kopię pakietu, w którym program ten został użyty. To wszystko.

Możliwości programu:

- odczytuje teksty spakowane PowerPackerem (jeśli tylko na dysku obecna jest biblioteka powerpacker.library),
- do wybierania tekstu do odczytania używa requesterów z bibliotek: reqtools.library, req.library lub arp.library (w zależności od tego, która jest dostępna) lub pobiera nazwę zbioru procedurą GetString,
- dość szybki scroll w obie strony,
- tryb Iconify, dzięki czemu może być stosowany jako rezydentny, podręczny program odczytujący teksty,
- automatycznie wyłącza wskaźnik myszy, żeby nie przeszkadzał w czytaniu (tak jak w CEDzie),
- jest bardzo krótki,
- pracuje na własnym ekranie o rozmiarach takich samych jak ekran Workbencha,
- używa dowolnej nieproporcjonalnej czcionki o rozmiarach 8x8,
- potrafi wyświetlać teksty w dowolnym standardzie kodowania polskich liter i co się z tym wiąże, jest bardzo dobrym narzędziem do konwersji tekstów ASCII pomiędzy różnymi standardami kodowania polskich liter.

Uruchomienie

Składnia przy uruchamiania z CLI wygląda następująco:

PolView -rozkaz NazwaTekstu

Rozpoznawane rozkazy:

-f NazwaFontu

Określa nazwę fontu który zostanie użyty do odczytywania tekstu. NazwaFontu to jego nazwa bez końcówki ".font" np. *TopazxJP, RubyxJP, PearlxJP*. Font musi być wielkości 8x8 i nie może być fontem proporcjonalnym. Jeśli dany font nie zostanie otworzony, użyty zostanie standardowy *Topaz*.

NazwaTekstu - nazwa tekstu do wczytania. Jeśli tekst nie zostanie wczytany to program przejdzie w tryb Iconify. Jeśli nie podaliśmy żadnej nazwy, to otworzy się requester z biblioteki reqtools.library/req.library/arp.library (w zależności od tego, która biblioteka będzie dostępna) i będziemy mogli własnoręcznie wybrać tekst do przeczytania.

Przykładowe wywołanie:

PolView -f TopazxJP s/startup-sequence

Z WorkBencha można dwukrotnie nacisnąć lewym przyciskiem myszy na ikonie programu. Otworzy się wtedy requester i będziemy mogli własnoręcznie wybrać tekst do przeczytania. Można także nacisnąć raz myszą na ikonie tekstu do odczytania, przytrzymać klawisz *SHIFT* i dwukrotnie nacisnąć na ikonie programu. Spowoduje to odczytanie wybranego tekstu.

PolView można także używać jako domyślnego programu do odczytywania tekstów. Wystarczy wtedy nacisnąć raz na ikonie tekstu, wybrać z menu Workbrncha pozycję Info. Pokaże się wtedy okno z informacjami o tym tekście. Do pola Default Tool wpisujemy nazwę PolView wraz z pełną ścieżką dostępu do programu (np. jeśli umieściliśmy program PolView w katalogu c to wpiszemy tam: "*:c/PolView*" (bez nawiasów oczywiście) i naciskamy "*Save*".

Program rozpoznaje następujące *ToolTypes* (wpisujemy tak jak *Default Tool*, tylko do pola *ToolTypes*):

FONT=NazwaFontu

"FONT" koniecznie musi być dużymi literami.

NazwaFontu - patrz wyżej.

Ponieważ program posiada możliwość konwersji tekstu i podczas wyświetlania na bieżąco dokonuje konwersji, aby dopasować tekst do aktualnego fontu, możemy także podać w jakim standardzie kodowania polskich liter jest:

- czcionka, którą używamy do wyświetlania,
- tekst, który został wczytany,
- na jaki standard będziemy chcieli przekonwertować tekst.

Możemy więc podać:

FONTSTD=nazwa standardu
SOURCESTD=nazwa standardu
DESTSTD=nazwa standardu

FONTSTD to standard czcionki, *SOURCESTD* to standard wczytywanego tekstu, a *DESTSTD* to standard na jaki chcemy (ewentualnie) przekonwertować tekst. Ustawienia te możemy zmieniać na bieżąco w programie.

Dostępnymi standardami są: Brak, XJP, AmigaPL, Mazovia, Latin-2, DHN Cyfromat, Ventura, CSK, Microvex

Brak - oznacza brak polskich liter.

Informację tą można przypisać także każdemu tekstowi odczytywanemu przez PolView. Wtedy dane zawarte w ikonie tektu są ważniejsze (jeśli np. do programu PolView wpisaliśmy dane: "*FONT=RubyxJP*", a do tekstu "*FONT =TopazxJP*", to jeśli wczytamy tekst przez dwukliknięcie na jego ikonie, to użyty zostanie font *TopazxJP*, natomiast gdy uruchomimy sam program, użyty zostanie *RubyxJP*.

Kiedy już załadujemy tekst, program otworzy własny ekran i wyświetli plik tekstowy. Na dole ekranu znajduje się linia statusu z kilkoma gadgetami służącymi do przewijania tekstu oraz informacjami takimi jak: nazwa tekstu, jego wielkość, ilość linii tekstu i numery linii widocznych na ekranie. Jeśli chcemy by tekst zajmował całą powierzchnię ekranu możemy użyć klawisza DEL, który przełącza między wyświetlaniem z linią informacyjną i bez niej. Tekst



można przewijać klawiszami kursora (o linię w górę, dół, w prawo i w lewo). Klawisze kursora z klawiszem *SHIFT* to szybkie przewijanie o stronę w górę/dół. Można też użyć do przewijania gadżetów lub klawiatury numerycznej (strzałki góra/dół o linie, PgUp/PgDn-strona w górę dół, Home-początek tekstu, End-koniec tekstu). Naciśnięcie:

SPCJA - przewijanie tekstu o stronę w dół,
BACKSPACE - przewijanie tekstu o stronę w górę,
HELP, "h", "?" - wyświetla spis dostępnych klawiszy,
ESC - wyjście z programu,
"l" - wczytywanie nowego tekstu do odczytania,
"i" - przechodzi w tryb iconify nie tracąc tekstu,
"I" - to samo co "i" tylko usuwa tekst z pamięci,
"s" - umożliwia zmianę standardów,
"n" - nagrywa przekonwertowany, na inny standard kodowania polskich znaków, tekst.

Tryb iconify polega na tym, że program zamyka swój ekran i otwiera na ekranie workbencha małe okno. Początkowo jest nim wyświetlany tekst z opisem używanych klawiszy. Możemy je zmniejszyć naciskając prawy przycisk myszy. Powrót do większego okna też prowadzi przez prawy przycisk myszy. Klawisze dostępne w trybie iconify:

"P" - powróć do wyświetlania tekstu,
"L" - załaduj nowy tekst i automatycznie go odczytaj,
"U" - usuń tekst z pamięci,
RMB - czyli prawy przycisk mysz: przełączanie między małym, a dużym oknem.

Program ułatwia wyświetlanie polskich tekstów w dowolnym standardzie kodowania polskich liter. Aby efektywnie korzystać z tej opcji musimy wiedzieć, w jakim standardzie polskich znaków została spolszczona czcionka i w jakim standardzie został napisany tekst. Program podczas wyświetlania dokonuje konwersji ze standardu tekstu na standard czcionki. Ustawienia te możemy zmienić naciskając klawisz "s". Ukaże się wtedy okno ze spisem dostępnych standardów (aktualnie jest ich 10) i pięcioma gadżetami nazwanymi: "FONT", "TEKST ŹRÓDŁOWY", "TEKST DOCELOWY", "AKCEPTUJ ZMIANY" i "ZANIECHAJ ZMIAN". Aby określić standard fontu, tekstu źródłowego lub tekstu docelowego wystarczy kliknąć na odpowiednim gadgecie. Na górze ekranu pojawi się informacja o tym, jaki standard jest aktualnie przypisany temu elementowi. Aby zmienić standard wystarczy nacisnąć na jego nazwie. Jeśli dokonaliśmy zmian możemy nacisnąć na "*AKCEPTUJ ZMIANY*". Jeśli nie chcemy zmian to naciskamy na "*ZANIECHAJ ZMIAN*" lub zamykamy okno.

W celu dokonania konwersji tekstu musimy przede wszystkim wiedzieć w jakim standardzie został pierwotnie napisany. Ustawiamy ten standard dla "*TEKSTU ŹRÓDŁOWEGO*". Teraz dla "*TEKSTU DOCELOWEGO*" ustawiamy standard, na który chcemy dokonać konwersji. Naciskamy "*AKCEPTUJ ZMIANY*" i następnie naciskamy klawisz "n". Wybieramy nazwę pod jaką chcemy nagrać przekonwertowany tekst i to wszystko. Na czas nagrywania tekstu wskaźnik myszy przyjmuje postać dyskietki.

Autorem programu jest Krzysztof Kowalczyk.

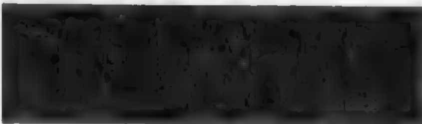
5. CheckSys

To prosty programik, po którego uruchomieniu uzyskujemy wyczerpującą informację na temat konfiguracji naszej Amigi. Uruchomienie możliwe jest jedynie z poziomu CLI i odbywa się poprzez wpisanie i zaakceptowanie nazwy programu. Dokładny opis tego narzędzia został zamieszczony w poprzednim numerze Amigowca.

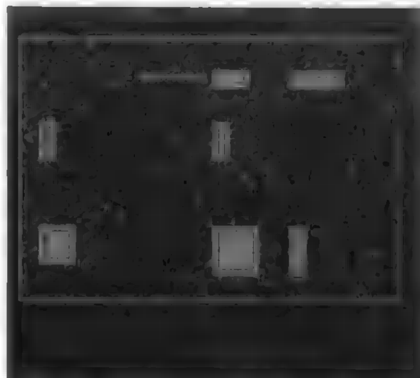
6. Programy

W katalogu o tej nazwie znajdują się listingi wszystkich przykładowych programów zamieszczonych w bieżącym numerze naszego pisma. Wszystkie programy źródłowe, głównie ze względów oszczędnościowych, nie posiadają ikonek oraz dodatkowo zostały spakowane programem PowerPacker 4.0. Dlatego przed ich wczytaniem pod edytor, należy bezwzględnie dokonać dekompresji danych. Można to zrobić przy pomocy PowerPackera lub programiku PP, który znajduje się na Amigowym dysku Public Domain #2/93.

7. The Worms



Jest to program, który można zaliczyć do ogromnej grupy gier zręcznościowo-logicznych (ze zdecydowaną przewagą elementów zręcznościowych). Opis zasad gry znajduje się w programie (HELP). Ogólnie mówiąc - sterując jednym z dwóch stale rosnących robaków należy zagonić przeciwnika w "kozi róg". Każde zetknięcie z przeszkodą lub przeciwnikiem kończy się dla zwierzaka tragicznie. Przed rozpoczęciem gry wybrać możesz ilość graczy, sposób kontroli, szybkość gry i stopień zagęszczenia przeszkód.



Program uruchomić można zarówno z WorkBencha jak i CLI.

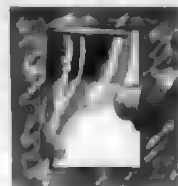
Autorem programu jest Rafał Mantiuk.

Autorem większości ikonek zamieszczanych na naszych dyskietkach jest Dariusz Zwierzyński (Animal).

Prosimy o uwagi o naszej bibliotece. Piszcie, do nas jakich programów poszukujecie i jakie powinny się w niej znaleźć. Czekamy też na owoce Waszej pracy.

Jak zamawiać i ile to kosztuje?

Dyskietki PD można zamawiać przesyłając pieniądze na nasze konto przy pomocy blankietu z Amigowca, podając symbol dysku PD (od #1/92 do #12/92, #1/93, #2/93, #3-4/93, #5-6/93, #7-8/93 oraz "A", "B" i "C") i oczywiście swój czytelny adres. Nasza dyskietka kosztuje 35 tysięcy złotych (wliczone są koszty dyskietki, przesyłki, nalepki, opakowania, itp.). W przypadku przesyłki za zaliczeniem pocztowym zamawiający pokrywa koszty zaliczenia. ❑



Lista demonów

Artur Łukasik, Sebastian Klomski

Witamy po raz kolejny! Na scenie zachodniej kilka znacznych zmian. W Polsce także coś się ruszyło po PARTY w Poznaniu.

ZAGRANICA
GRUPY:

1. **Sanity**
2. **Melon Dezign**
3. **Kefrens**
4. **Lemon**
5. **Silents**
6. **Andromeda**
7. **Spaceballs**
8. **Complex**
9. **Dual Crew - Shining**
10. **Movement**

Grupa Melon spadła na drugie miejsce. Jest to zapewne spowodowane zmianą stylu. Ich nowe demka składają się ze digitalizowanych obrazków oraz samplowanej muzyki. Z pewnością dema te miały rozśmieszać, jednak wiele osób uważa je za najgorsze w historii sceny. Miejmy nadzieję, że Melon powróci do swojego dawnego stylu jaki mieliśmy okazję oglądać w *How to Skin a Cat.*

DEMA:

1. ***Hardwired*/Cronics & The Silents**
2. ***Desert Dream*/Kefrens**
3. ***Extension*/Pygmy Projects**
4. ***How to Skin* a Cat/Melon Dezign**
5. ***Groovy*/Lemon**
6. ***D.O.S.*/Andromeda**
7. ***State of The Art*/Spaceballs**
8. ***D.A.N.E.*/Kefrens**
9. ***World of Commodore*/Sanity**
10. ***Guardian Dragon II*/Kefrens**

Na pozycji trzeciej pojawiło się demko ***EXTENSION*** nowej grupy o nazwie ***Pygmy Projects***. Grupę tę założyli uchodźcy z **The Silents**, znani wcześniej z ***Maximum Velocity***. Demo to nie było rewelacją, toteż tym bardziej dziwi fakt wypuszczenia produktu tak dobrego jak ***Extension***. ***Extension*** zaskakuje nas rewelacyjną muzyką autorstwa ***Jestera/Sanity*** oraz kilkoma nowymi procedurami. Gorzej natomiast z grafiką oraz z designem.

DEMA PLIKOWE:

1. ***Interference*/Sanity**
2. ***Mindriot*/Andromeda**
3. ***3D Demo II*/Anarchy**
4. ***Announce*/Lemon**
5. ***Transpiersky*/The Silents**
6. ***Booo*/Melon Dezign**
7. ***Terminal Fuckup*/Sanity**
8. ***Optimum Fuckup*/Sanity**
9. ***Joyride*/Phenomena**
10. ***Hardcore*/Anarchy**

INTRA 40 KB:

1. ***Tetris*/Melon Dezign**
2. ***Lemon***
3. ***Shinig***
4. ***Fairlight***
5. ***Arrilfoll*/Dual Crew & Shinig**
6. ***No Pain No Gain*/TRSI**
7. ***Stellar***
8. ***The Silents***
9. ***TEK***
10. ***Scoopex***

Pojawiło się ostatnio dość dużo nowych interek. Niestety, większość z nich stoi na dość niskim poziomie, tak więc intro Tetris grupy Melon Dezign nie znalazło konkurenta zdolnego zrzucić je na niższe pozycje.

MUSIC DISKI:

1. ***Jesterday*/Sanity**
2. ***Mirror*/Andromeda**
3. ***Memorial Songs*/Alcatraz**
4. ***Crystal Symphonies II* /Phenomena**
5. ***Turmoil*/Sanity**
6. ***Kuglep0len*/Static Bytes**
7. ***Book of Song*/Complex**
8. ***Music Dream II*/Phenomena**
9. ***Imperial Tunes 2*/Parasite**
10. ***Crystal Symphonies I* /Pheno mena**

KODERZY:

1. ***Chaos*/*Sanity***
2. ***The Spy*/Crionics**
3. ***Hannibal*/Lemon**
4. ***Laxity*/Kefrens**
5. ***Slammer*/Melon Dezign**
6. ***Lone Starr*/Spaceballs**
7. ***Microforce*/Sanity**
8. ***Bannasoft*/Melon Dezign**
9. ***Dr.Jekyll & Mr.Hyde* /Andromeda**
10. ***Performer*/Melon Dezign**

Chaos jak zwykle na topie. Właściwie żaden z aktywnych obecnie koderów nie jest w stanie mu zagrozić. Tylko powrót The Spy mógłby zmienić tę sytuację.

GRAFICY:

1. ***Facet*/Lemon**
2. ***R.W.O.*/Kefrens**
3. ***Cougar*/Sanity**
4. ***Peachy*/TRSI**
5. ***Mack*/Melon Dezign**
6. ***Fairfax*/Andromeda**
7. ***UNO*/Scoopex**
8. ***Walt*/Melon Dezign**
9. ***Havok*/Sanity**
10. ***J.A.D.E.*/Lemon**

R.W.O. nieustannie pnie się do góry. Jeszcze kilka demek z jego grafikami i zajmie miejsce równie dobrego ***Faceta***.



Natomiast ***Peachy*** zaczyna spadać, [illegible] jego pozycje zajmują inni bardzo zdolni graficy.

MUZYCY:

1. ***Jester***/**Sanity**
2. ***Audiomonster***/**Melon Dezign**
3. ***Lizard King***/**Alcatraz**
4. ***Laxity***/**Kefrens**
5. ***Romeo Knight***/**TRSI**
6. ***Tip [illegible] Mantronix***/**Phenomena**
7. ***4-mat***/**Anarchy**
8. ***Bit Arts***/**Complex**
9. ***Jesper Kyd***/**The Silents**
10. ***Virgill***/**Masque**

Jester bezapelacyjnie na samej górze. Audiomonster zrobił się ostatnio mało produktywny i musiałby się nieźle pomęczyć, aby zrzucić Jestera.

POLSKA
GRUPY:

1. ***TILT***/**Old Bulls**
2. **Mad Elks**
3. **Union**
4. **Deform**
5. **Alchemy**
6. **W.F.M.H.**
7. **Applause**
8. **Joker**
9. **Suspect**
10. **Investation**

Na pierwszym miejscu grupa TILT, która wygrała konkurs na najlepsze demo na POLISH AUTUMN'93. Zgodnie z naszymi oczekiwaniami grupa Applause pnie się do góry. Co prawda pomału, ale jeszcze jedno lub dwa demka z rewelacyjną grafiką TPP i kto wie? Warty odnotowania jest także fakt powrotu [illegible] scenę grupy Deform. Otóż jej główny koder po odejściu od Union wystawił na PARTY w Poznaniu rewelacyjne demko. Demko to niestety nie znajdzie się na naszej liście ponieważ zostało zrobione tylko na Amigę 1200, a takiej rybryki na razie nie mamy. Na pozostałych miejscach w zasadzie bez większych zmian.

DEMA:

1. ***Impulse***/**TILT**
2. ***Technological Death*** /**Mad Elks**
3. ***Fugazi***/**Old Bulls (TILT)**
4. ***Deformations***/**Deform**
5. ***Software***/**W.F.M.H.**
6. ***Marchewki***/**Alchemy**
7. ***Hallucinations [illegible] Dreams*** /**Union**
8. ***The Return***/**Joker**
9. ***Uquala***/**Damage**
10. ***PRO.s.l.a.k***/FCI

Na pozycji pierwszej rewelacyjne demko grupy TILT - Impulse. Ci, którzy je widzieli wiedzą, że zasługuje ono na pierwsze miejsce. Na Party w Poznaniu pojawiło się także kilka innych produkcji (najlepsze z nich na pozycji dziewiątej i dziesiątej), jednak ich poziom nie nominuje żadnej z nich wyższe pozycje.

DEMA PLIKOWE:

1. ***Party-zanci***/**Joker**
2. [illegible] ***Bara***/**Applause**
3. ***Impulse-preview*** /**Old Bulls (TILT)**
4. ***Varathron***/**Suspect**
5. ***Anam***/**Investation**
6. ***Autobiography***/**Old Bulls**
7. ***Xenium***/**Old Bulls**
8. ***Ingis***/**Investation**
9. ***Defene coś tam***/**Luzers**
10. ***Optical Race II***/**Investation**

W demach plikowych zastój. Żadna z dobrych polskich grup nie wypuściła demka plikowego.

DYSKI MUZYCZNE:

1. ***Vengeance***/**Beta Team**
2. ***Orgasm***/**Old Bulls (TILT)**
3. ***Waiting for Another Beer*** /**Joker**
4. ***Kadi Sound Disk III***/**Joker**
5. ***Music Tracks***/**Suspect**
6. ***Illusion***/**Old Bulls**
7. ***Hunt for the red kret***/**FCI**
8. ***Kadi Sound Disk II***/**Joker**
9. ***Techno Fuss***/**Investation**
10. ***Sigh***/**Old Bulls**

Nowy sound disk Petersa trochę rozczarował, tzn. muzyka jest na dobrym poziomie jednak jego oprawa pozostawia wiele do życzenia. Stąd tylko trzecia pozycja. Ostatnio na polskiej scenie można zauważyć dziwne zjawisko. Otóż dyski muzyczne ze świetną muzyką mają raczej niezbyt ciekawą oprawę (Vengeance, Waiting for Another Beer) i na odwrót. Ciekawe dlaczego?

KODERZY:

1. ***Musashi***/**Union**
2. ***Hudi***/**TILT**
3. ***Bukka***/**TILT**
4. ***Robin***/**W.F.M.H.**
5. ***Vico***/**Alchemy**
6. ***Dak***/**Mad Elks**
7. ***Tom***/**Union**
8. ***Kane***/**Suspect**
9. ***GBH***/**Joker**
10. ***CBM80***/**Beta Team**

MUZYCY:

1. ***XTD***/**Union**
2. ***Peters***/**Joker**
3. ***Pic Saint Loup***
4. ***Mr.Root***/**Union**
5. ***Snoopy***/**Union**
6. ***Accord***/**Gel Dezign**
7. ***Dreamer***/**FCI**
8. ***Passat***/**Funzine**
9. ***Seq***/**Union**
10. ***Kadi***/**Joker**

GRAFICY:

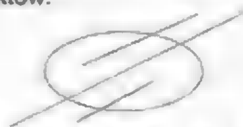
1. ***Animal***/**Union**
2. ***Pluton***/**Union**
3. ***TPP***/**Applause**
4. ***Sabe***/**TILT**
5. ***Seq***/**Union**
6. ***Python***/**Union**
7. ***Rys***/**Joker**
8. ***Berserker***/**Investation**
9. ***Frodo***/**Alchemy**
10. ***Katarek***/**Turnips**

MAGAZYNY DYSKOWE:

1. ***Zig Zag***/**Union**
2. ***Paper White***/**Luzers**
3. ***Zasmashka***/**Alchemy**
4. ***Fat Agnus***/**Investation**
5. ***Thing***/**Applause**
6. ***Imazine***/**Saint Group**
7. ***Poczytaj Mi Mamo***
8. ***X-mag***/**Union**
9. ***Przypadkownik***/**Generation**
10. ***Krawężnik***/**Status OK**

Wszystkie wiadomości pochodzą z pewnych źródeł i są w 100 procentach sprawdzone.

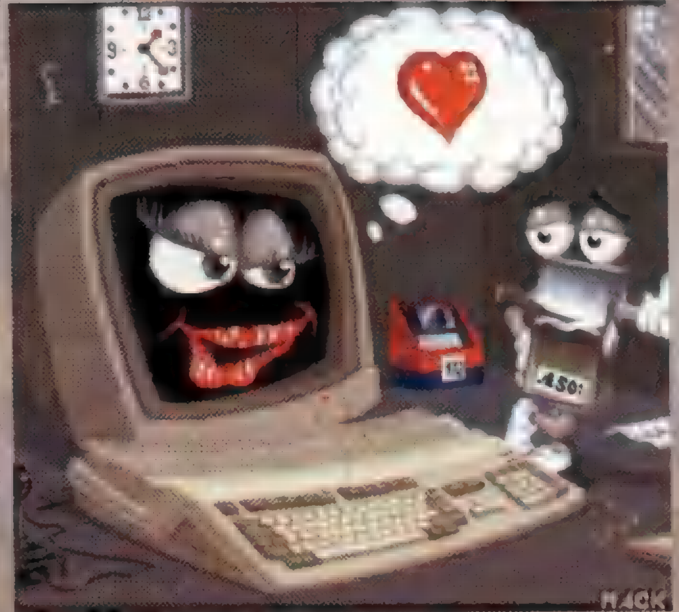
Głosujemy na dowolną ilość pozycji z każdego działu przyznając im od 1 do 10 punktów.



Głosy, propozycje, swoje prace, uwagi, a także najnowsze programy demonstracyjne i magazyny dyskowe prosimy przesyłać na adres redakcji.❑




HEY YOU !
DRINK ME...


ASO!
HACK


MAD
ELKS









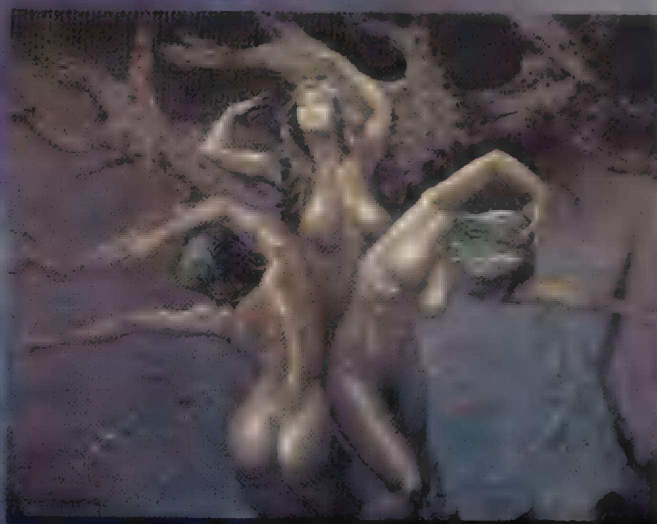

MAD
ELKS

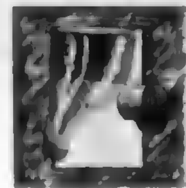







melon.





Polish Autumn '93

Marcin [redacted]wski

W dniach 23-24 października 1993 roku odbyło się w Poznaniu amigowe Party zorganizowane przez miejscowe grupy (Mad Elks, Flying Cows Inc, Dioxide, Lesiu Ind.). Poznańskie Technikum Energetyczne, w którym impreza miała miejsce, stało się na dwa dni stolicą krajowej sceny.

Po zapłaceniu [redacted] tysięcy, każdy otrzymywał okolicznościowy identyfikator oraz dwie karty do głosowania. Jedną na sobotę (music, graphic i demo competition) i jedną na niedzielę (intro compo). Istniała możliwość wynajęcia [redacted] darmo sali z kluczem (po wpłaceniu 500-tysięcznej kaucji, która miała zapobiec dewastacji), jednak dla niektórych grup zabrakło pomieszczeń (nawet po uprzedniej rezerwacji). Często w jednej sali rezydowało po parę grup. Party przebiegało w spokojnej atmosferze. Jedynym niemiłym akcentem był fakt podpalenia firanki w jednej z sal, załatwiony polubownie (przepadła część kaucji).

Dużym plusem był bufet czynny przez cały czas trwania imprezy. Można było zjeść i wypić bez potrzeby wychodzenia z party. Mimo to co jakiś czas wyruszały ekipy do pobliskich sklepów całodobowych (ciekawe po co?).

Sala, w której odbywały się konkursy nie była ogromna i z ledwością mieściła wszystkich uczestników. Dzięki temu atmosfera była właściwa (szczególnie podczas demo-compo).

Wszystkim zaimponowały też nagrody, po raz pierwszy wręczane nie tylko za pierwsze miejsca, ale także za drugie i trzecie. Podział wyglądał następująco:

music, graphic, intro competition:
1 miejsce - 1.500.000 zł
2 miejsce - 750.000 zł
3 miejsce - 350.000 zł

demo copetition:
1 miejsce - 3.000.000 zł
2 miejsce - 1.500.000 zł
3 miejsce - 750.000 zł

Jak to zwykle na party bywa, pierwsze odbyło się music-compo, poprzedzone selekcją (z ponad pięćdziesięciu modułów wybrano dziesięć). Poziom wyselekcjonowanych kompozycji był naprawdę wysoki i wyrównany. Oto wyniki:
1 - LIGHT & SOUND 4 BY: DREAMER
2 - TRANSFIX BY: XTD
3 - GRUMPLE BY: SCORPIK
4 - ENFORCER BY: DAN
5 - WHAT IS IT BY: ROBERTS
6 - UNPLUGGED BY: ACCORD
7 - UTOPIA ISLAND BY: PETERS
8 - SLAVE OF TECHNO BY: MICHAL R.
9 - METROPOLIS BY: WIERZA
10 - KILL THE HATE BY: PAVELO

Następny w kolejności był konkurs na najlepszą grafikę. Każda osoba mogła wystawić trzy obrazki. Nic więc dziwnego, że uzbierało się ich ponad 120. Ray-tracingi oceniano oddzielnie. Poziom jak zwykle był ogromnie zróżnicowany. Wygrał SEQ, [redacted] dziwić może odległa pozycja najlepszego dotychczas polskiego grafika - Animala. Oto pierwsza dziesiątka najładniejszych grafik:
1 - BLACK JACK BY: SEQ
2 - OMEGA BY: 997
3 - MISS KISS BY: TPP
4 - NATURA BY: PLUTON
5 - MOONLIGHT BY: MR MELON
6 - COLUMB BY: EDI, BAILEYS BY: ?
7 - VALESKO BY: ?
[redacted] - ZMROK BY: PLUTON
9 - BADYLE BY: XENOS
10 - FLANTA BY: MARCIN BOGASZ

Kulminacją wieczoru było demo-competition. Zgłoszono cztery dema na Amigę 500 i jedno na Amigę 1200. W kategorii A500 zwyciężyło ładne demko *Impulse* grupy ***TILT***, a w kategorii A1200 [redacted] powodu braku konkurencji niesamowity produkt *Musashiego* wydany pod szyldem ***Investation***. Organizatorzy uhonorowali ten produkt oddzielną nagrodą. Po demo-compo wyniknął mały skandal, ponieważ zniknęły dyski z demem Tiltu, a że była to jedyna kompletna kopia, nikt nie zdążył sobie tego dema przegrać. Zresztą demo działało tylko z keylock'iem wkładanym w port joystick'a, więc i tak nikt by go sobie nie obejrzał. Trochę to nieładnie ze strony autorów, że przyjęli nagrodę za demo którego praktycznie nie ma. Wszyscy czekają więc na normalną wersję Impulse'a. Kolejność dem była następująca:

Kategoria Amiga 500:

1. Impulse/Tilt 3429 pkt
2. Pro-siak/Flying Cows Inc. 2832 pkt
3. Uquala/Damage 2033 pkt
4. Overvision/Blaze 1061 pkt

Kategoria Amiga 1200:

1. Rayworld/Deform+Investation

Po ostatnim z sobotnich konkursów większość zajęła się swoimi sprawami. Inni poszli się zabawić, [redacted] jeszcze inni z własnej woli (lub też z konieczności) udali się na zasłużony odpoczynek.

W niedzielę rano odbył się konkurs na najlepsze intro o długości nie przekraczającej 40KB. Zwyciężyła grupa Suspect z produkcją w swoim stylu.

Około południa odbyło się uroczyste wręczenie nagród dla najlepszych. Były to naprawdę przyjemne chwile w porównaniu z podobnymi "uroczystościami" na innych party.

Impreza była zorganizowana profesjonalnie i w iście europejskim stylu. Organizatorzy wywiązali się znakomicie ze swojego zadania i nic poważnego nie można im zarzucić. Myślę więc, że [redacted] pamięci uczestników, Polish Autumn'93 pozostanie jedną [redacted] najlepszych imprez amigowych jakie ostatnio zorganizowano w naszym kraju. ❑





producent: ASDG
dystrybutor: MacroSystem Computer GmbH
Friedrich Ebert Str. 85
58454 Witten
Niemcy
zgodność: 2.x, 3.x
cena: 400 DM

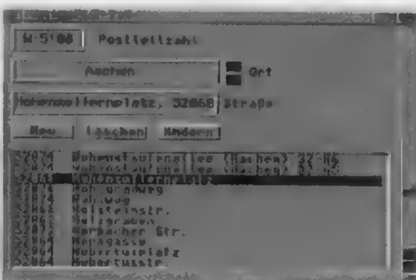
ADPro 2.3

ArtDepartment Professional jest programem, który znają wszyscy zajmujący się obróbką grafiki komputerowej. Program ten pozwala na przetwarzanie grafik. Można przy jego pomocy zmieniać wielkość rysunku, ilość kolorów, przetwarzać negatyw na pozytyw, retuszować pojedyncze kolory, itp. Dodatkowe moduły umożliwiają np. skanowanie. Program ma bezpośrednie połączenie np. z ProPagem. Jest to moloch niezbędny w każdym studiu graficznym. Jedyną jego wadą jest zapotrzebowanie na ooooogromne ilości pamięci. Czasami niewystarcza nawet pełne rozszerzenie pamięci Amigi do 18 MB.

Ostatnio pojawiła się [redacted] wersja tego programu oznaczona 2.3. Co nowego? Przede wszystkim zmieniono panel obsługi. Pozostawiono kolory i styl wyglądu, ale pozmieniano pozycje wywołań poszczególnych funkcji. Teraz są one w pełni kompatybilne z programem Morph Plus. Poza tym dodano możliwość współpracy z ARexxem, co z pewnością zaowocuje pojawieniem się nowych Geni dla ProPage'a i pełniejszym wykorzystaniem ADPro przez inne programy graficzne. Wprowadzono też możliwość pełnej obsługi programu z klawiatury.

ADPro 2.3 ma możliwość bezpośredniej wymiany danych z programem DPaint. Niestety tylko z najnowszą amerykańską wersją, której założono odpowiednią "łatę".

Odnowiono działanie driverów określanych w programie jako "Operators", "Loader" i "Saver". To one zawiadują funkcjami programu i dzięki nim można ładować czy też eksportować grafiki w rozmaitych standardach, z których wiele jest nowymi "odkryciami". Między innymi możliwe jest zgrywanie animacji w systemie umożliwiającym szybsze działanie [redacted] komputerach 32-bitowych.



Producent: OASE Software
dystrybutor:
OASE Software
Schurkamp 24,
48720 Rosendahl-Osterwick
Niemcy
zgodność: 1.3, 2.x, 3.x,
cena: 70DM

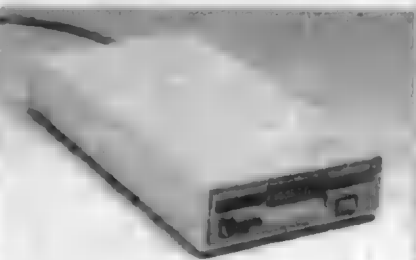
Nowa książka kodowa

Wszystkim, którzy prowadzą szeroko zakrojoną korespondecję z firami w RFN gorąco polecamy procukt firmy Oasa "Amiga-PLZ". Jest to komputerowa książka kodowa dla Niemiec. Zestaw składa się z 5 dyskietek zaś instalacja na twardym dysku następuje poprzez poprzeciąganie 5 kolejnych ikonek do odpowiedniej szuflady na dysku.

Program umożliwia wyszukiwanie nowych numerów kodowych według starych numerów, miejscowości i ulicy. Wystarczy podać np. nazwę miasta i ulicy i już można wysyłać list do znajomego. Oczywiście stare numery kodowe podaje się z przedrostkami "W-" dla byłych Niemiec Zachodnich lub "O-" dla byłych Niemiec Wschodnich. Poniżej jest okienko do wpisywania nazwy miasta. Jeśli ktoś nie jest pewien jej brzmienia może sobie poprzeglądać nazwy za pomocą strzałek z boku okienka. Można także wszystkie nazwy wydrukować. Niestety drukowana jest cała lista, a zatrzymanie wydruku jest możliwe jedynie przez wyłączenie drukarki lub/i zresetowanie komputera. Jeszcze poniżej pojawia się pole tekstowe, z którego możemy wybrać ulicę po prostu poprzez kliknięcie. Jeśli dotkniemy jakiejś litery na klawiaturze [redacted] oknie pojwią się nazwy zaczynające się właśnie od niej.

Istnieje możliwość wprowadzania własnych modyfikacji do programu polgających na podaniu nowego numeru kodowego, nowej nazwy ulicy, itp.).

Pomyślano również [redacted] użytkownikach z większych firm i program ma możliwość wykorzystania portu ARexxa. Teraz np. [redacted] stworzyć krótki programik, który pozamienia nam stare numery kodowe w adresach korespondentów na nowe.



producent: Power Computing Ltd.
dystrybutor:
GTI
Zimmersmühlenweg 73,
61440 Oberursel
Niemcy
zgodność: 2.x, 3.x
cena: 300 DM

HD dla Amigi

Nowe Amigi (nie wszystkie) 3000 i 4000 są standardowo wyposażane w stacje dysków o dużej gęstości. Powstaje problem, gdy nasza "mała" przyjaciółka służy nam do współpracy z IBMem lub gdy mamy do magazynowania większe ilości danych. Oczywiście należy kupić sobie HD! Podobno w Amigach 1200 stacja taka miała być montowana standardowo. Okazało się jednak, że ich obudowa jest o kilka milimetrów za niska i stacje HD firm, u których zaopatruje się Commodore nijak tam nie pasują. No cóż złośliwość losu. Oczywiście istnieje cała masa zewnętrznych stacji HD do Amigi. Niektórzy nawet radzą wywalenie stacji wewnętrznej i wstawienie [redacted] jej miejsce dysku twardego 3.5 cala i korzystanie ze stacji zewnętrznej, jak i [redacted] wewnętrznej.

Stację HD 3.5 calową można podłączyć do Amig wyposażonych [redacted] Kickstart od 2.0 w górę. Na rynku spotka się wprawdzie wiele stacji tego typu, jednak działają one niezgodnie ze standardami technicznymi i raczej trudno przy ich pomocy odczytywać dyski MS-DOSa. Sprawę rozwiązuje stacja PowerDrive firmy Power Computing Ltd. kosztująca ponad 300 DM. Stacja umożliwia współpracę zarówno z dyskietkami HD jak i normalnymi DD. Od systemu 3.0 istnieje możliwość pracy na dyskach MS-DOSowychpoprzez systemowy program CrossDos.

Stacja dostarczana jest z dyskietką instalacyjną, na której znajduje się programik odpowiednio ustawiający system. Współpra[redacted] zprogramikiem CrossDos moze początkowo przysporzyć nieznacznych problemów.

Podsumowując można powiedzieć, ze cena stacji jest wysoka i nie wiadomo komu kiedy się zwróci. Warto zrobić dokładniejszy bilans.



"Żarty" z dyskiem twardym

Tomasz Kokoszczyński

W ostatnim AMIGOWCU na stronie 38 ukazał się artykuł Twardy dysk dla Amigi. Chodziło oczywiście o Amigę 1200. Mottem przewodnim artykułu było zamontowanie dysku 3.5 calowego zamiast dysku 2.5 calowego. Okazuje się, że można to przeprowadzić [redacted] mniej bolesny sposób...

W artykule autorzy sugerują, że trzeba sporo [redacted] naszej Amigi powycinać by uzyskać miejsce na umieszczenie twardego dysku 3.5 cala. Wycina się fragment ożebrowania, wierci z tyłu [redacted] dziury i usuwa cały metalowy ekran. W momencie gdy dokonywali oni tego montażu faktycznie była to jedyna znana metoda i proszę nie mieć im tego [redacted] złe, że mimowolnie wprowadzili Państwa w błąd. Cóż zdarza się. Z uwagi na szkody jakie taki sposób montażu mógłby wyrządzić waszej przyjaciółce postaraliśmy się jak najszybciej wydrukować sprostowanie.

Po [redacted] wiercić, piłować i usuwać, skoro można prościej, łatwiej i przyjemniej? Poza tym gdy dojdzie co do czego (tzn. do sprzedaży komputera) może okazać się, że nie ma chętnych na taką szpetotę.

W poszukiwaniach twardego dysku należy zdać się praktycznie [redacted] ślepy traf. Oczywiście nazwy "Conner", czy "Quantum" [redacted] bardzo dobrymi wskaźnikami. Sam jestem szczęśliwym posiadaczem A1200 z dyskiem "SeaGate". A tak między nami to okazało się, że jest to jeden z najszybszych dysków. Sysinfo szacuje jego szybkość na 1.3 Mega (sic!) na sekundę. Dla wielu programów szybkość ta jest i tak za duża.

U niektórych moich kolegów zdarzyło się, że przy niektórych twardych dyskach komputer wyłącza się po 15 minutach pracy. Wynika z tego, że zasilacz nie jest w stanie go zasilić i "wychodzi z siebie". Powrót następuje zazwyczaj po kilku godzinach. Marna perspektywa pracy [redacted] taką Amigą. Być może nie jest to spowodowane tym, że zasilacz jest za słaby, ale że gdzieś w podczas montowania powstało zwarcie, które powoduje zużycie części energii ponad miarę. Przy dobrym montażu zjawisko to nie powinno raczej wystąpić.

Jeśli chodzi o przejściówki z 2.5 na 3.5 cale to faktycznie należy postąpić tak jak radzą autorzy. Z tym, że złącza te są przeważnie wetknięte "na taśmę" (tzn. kabel). Wystarczy je po prostu zdjąć, bez potrzeby uciekania się do lutownicy, czy cięcia nożem.

UWAGA: złącze 3.5 calowe ma przeważnie szerszy rozstaw "szczęk", co wymaga odpowiedniego dociśnięcia ząbków, gdy włożymy w nie taśmę 2.5 calową! Jeśli tego nie zrobicie - twardy dysk może nie zaskoczyć. Tak właśnie było u mnie. Niby wszystko wyglądało cacy, ale system "bawił" się z twardym dyskiem i [redacted] koniec nie mógł go rozpoznać. Dociśnięcie ząbków do siebie dało natychmiastowy efekt (kolego, ale nie przy włączonym komputerze). Okazuje się, że w nowym sposobie montażu właśnie prawidłowe wykonanie złącza jest najtrudniejszą sprawą. Reszta to małe piwo z pianką.

Jak wpadliśmy na nowy sposób montażu dysku 3.5 calowego do A1200? Po prostu jeden kolega tłumaczył przez telefon drugiemu koledze jak to zrobić. Na linii nastąpiło niewielkie przekłamanie informacji. Potem ten kolega, który tłumaczył, lekko się zdziwił, gdy zobaczył Amigę, tego który montował. Nie było [redacted] niej śladu, że jest tam jakiś twardy dysk. Jedynie odpowiednia dioda migała przy pracy, no i słychać było warkot silnika twardziela.

Po pierwsze: niczego nie wycinamy.

Po drugie: no niestety trochę trzeba usunąć, bo jak pewnie zauważyliście A1200 jest komputerem małym, a nawet bardzo małym. W porównaniu z najmniejszymi z dużych, starszych braci typu IBOL A1200 wygląda jakby ktoś do klawiatury zapomniał czegoś dołączyć. No cóż, większy nie znaczy wcale lepszy.

UWAGA: Usuwamy z naszej A1200 dwie rzeczy: metalową podstawkę pod dysk 2.5 calowy oraz malutki kwadratowy ekranik, który jest przyłączony do ekranu właściwego.

UWAGA: najważniejszą czynnością jest odizolowanie układów elektronicznych twardego dysku od ekranu Amigi. Metalowa obudowa twardego dysku może bez problemu stykać się [redacted] metalowym ekranem. Nie mogą natomiast układy elektroniczne. Do odizolowania należy użyć płaskiego kawałka czegoś. Powinno to być dobrym izolatorem, a jednocześnie nie poddawać się łatwo wysokim temperaturom. [redacted] wytrzyma chwilę kontaktu [redacted] lutownicą to świetnie się nadaje jako podkładka. Przyklejamy ją od spodu twardego dysku.

Teraz możemy przystąpić [redacted] montażu. Okazuje się, że przewód łączący komputer z klawiaturą może iść górą zamiast dołem. Wystarczy potrzymać twardy dysk, włożyć przewód, docisnąć złącze i już możemy dokleić twardy dysk taśmą do ekranu. Taśma oczywiście nie powinna poddawać się tatwo topieniu. Radzę użyć dobrej firmowej taśmy np. taśmy ze wzmacniającymi włoskami firmy 3M. Taśma ta klei dobrze i nie da się jej rozerwać gołą ręką, nie mówiąc już o twardym dysku, no i [redacted] najważniejsze nie topi się zalewając nam drogocenne układy.

To praktycznie wszystkie zmiany. Resztę należy przeprowadzić tak jak podano [redacted] artykule.

Podsumowując:
Niczego nie wycinamy!

Po odłączeniu klawiatury usuwamy z komputera podstawkę pod dysk 2.5 calowy oraz malutki kwadratowy ekranik.

Dysk kładziemy w rynience utworzonej przez ekran (opieramy go o kancik).

Przed ułożeniem dysku podłączamy klawiaturę. Zielona taśma klawiatury idzie nad dyskiem i ma luz (niewielki, bo niewielki, ale ma).

Elementy elektroniczne dysku nie powinny stykać się z ekranem komputera. Metalowy bok dysku może dotykać ekranu.

Pozostałe łączenia wykonujemy jak to op[redacted] w artykule w poprzednim AMIGOWCU.

Dysk należy dobrze przykleić, ponieważ jego obluzowanie grozi zniszczeniem komputera i/lub dysku.

Dysk powoduje niewielkie naprężenie obudowy w rejonie klawiszy F5 i F6, które nie powinno mieć wpływu na pracę komputera.

Formatowanie dysku najlepiej wykonać z systemu. Radzę włączyć opcję "Directory Cache", która znacznie przyspiesza wczytywanie katalogu dysku.

Redakcja serdecznie przeprasza za wszelkie problemy, które wynikły przy montażu twardych dysków do Waszych Amig. Pamiętajcie jednak, że mieliśmy i mamy najszczersze chęci by Wam pomóc, [redacted] możliwość oszczędzenia kilku milionów na pewno do takiej pomocy należy. ❑



Amos Basic część 4
Tekst i okna

Tomasz Hrycuniak

Zgodnie z obietnicą daną w poprzednim odcinku, tym razem zajmę się ogólnie mówiąc tekstem. Opiszę podstawowe komendy przeznaczone do wyświetlania tekstu, sterowania oknami i zasady posługiwania się fontami dyskowymi.

Rozpocznę jednak od sposobu wprowadzenia polskich znaków do edytora amosa. Jest to konieczne, aby napisać jakiś program szerzej korzystający z tekstu. Operacja ta jest dosyć prosta, gdyż autor amosa przygotował do tego celu odpowiednie procedury, napisane rzecz jasna także w amosie. Pracę należy rozpocząć od przekopiowania do katalogu fonts dysku systemowego (dyskietki [redacted] której uruchamiamy system) fontu zawierającego polskie znaki. Najlepiej będzie użyć zestawu "tpzpl". Następnie uruchamiamy amosa i wczytujemy program Font_Convert.Amos. Pozwoli on na wykonanie konwersji fontu systemowego na standard amosa. Przetworzoną czcionkę zgrywamy bezpośrednio do katalogu Amos System (dyskietki [redacted] amosem) pod nazwą Default.font. Następnie trzeba uruchomić program Keyboard_Definer.Amos. Pozwoli on nam na przygotowanie odpowiedniej mapy klawiatury, czyli podstawienie polskich znaków "pod" odpowiednie klawisze. Przygotowaną mapę trzeba zgrać [redacted] katalogu Amos_System pod nazwą Default.Key. Obsługa obu wspomnianych programów jest bardzo łatwa i na pewno nie sprawi nikomu trudności. Gdy polskie znaki będą już w systemie, można przystąpić do pisania programu [redacted] ich wykorzystaniem. Przedstawię kilka komend związanych z operacjami [redacted] tekście.

Pen a
a - Kolor tekstu wyświetlanego w oknie bieżącym. Należy podać numer barwy. Ilość dostępnych kolorów zależy od trybu ekranu.
Zmiana koloru tekstu wyświetlanego [redacted] ekranie.

Paper a
a - Kolor [redacted] pod literami. Analogicznie do Pen.
Zmiana koloru tła pod literami.

Inverse ON / OFF
Zamiana kolorów tła i tekstu. ON włącza tę funkcję. [redacted] OFF wyłącza.

Under ON / OFF
Włącza lub wyłącza podkreślenie wyświetlanego tekstu.

Locate x,y
x,y - Współrzędne położenia kursora. Ustawienie kursora tekstowego w żądanym miejscu. Punkt o współrzędnych 0,0 znajduje się w lewym górnym rogu okna tekstowego.

Home
Umieszcza kursor w lewym górnym rogu okna. Równoznaczne z Locate 0,0.

Cdown
Przenosi kursor o jedną linię w dół.

Cup
Przenosi kursor o jedną linię w górę.

Cright
Przenosi kursor o jeden znak w prawo.

Cleft
Jak wyżej tylko, że w lewo.

Curs ON / OFF
Czyni kursor widocznym lub niewidocznym. Wprowadzanie i wyświetlanie tekstu działa poprawnie, tyle że kursor jest niewidoczny (po Curs OFF). Dotyczy to tylko bieżącego okna.

Curs Pen a
a - Kolor kursora.
Zmienia kolor kursora. Podstawowym jest 3, jedyny z flash'em (mrugający).

Centre a$
a$ - Zmienna tekstowa.
Działa analogicznie [redacted] Print z tym, że tekst wyświetla zawsze po środku aktualnej linii kursora.

Hscroll a
a=1 - Przesunięcie bieżącej linii w lewo.
a=2 - Przesunięcie całego tekstu w oknie w lewo.
a=3 - Jak przy "1" tylko w prawo.
a=4 - Jak przy "2" tylko w prawo.
Komenda ta umożliwia przesunięcie tekstu w oknie w poziomie.

Vscrol[redacted]
a=1 - Przesunięcie w dół tekstu z linii kursora i poniższych.
a=2 - Przesunięcie w górę tekstu [redacted] linii kursora i poniższych.
a=3 - Przesunięcie w górę tekstu [redacted] linii kursora i powyższych.
a=4 - Przesunięcie w dół tekstu z linii kursora i powyższych.
Dwie powyższe komendy są dosyć często stosowane do tworzenia efektu "płynącego tekstu" (scroll). Ich działanie najlepiej zilustruje przykład.

Prosty efekt "płynącego tekstu"

```
P#=4
Cdown
Centre "Płynący tekst CRTL+C to Stop"
Locate 0,20
Print "Prawy klawisz myszki - Przyspieszenie"
Print "Lewy klawisz myszki - Zwolnienie"
Wind Open 1,35,30,30,4,1
Curs off
Do
Locate 22,0
Print "AMIGA"
Home
For A=0 To 26
Wait P#
Hscroll 1
If Mouse Key=1 Then P#=P#+0.5
If Mouse Key=2 Then P#=P#-0.5
If P#<1 Then P#=1
Next A
Loop
```

Do tej pory wszystkie [redacted] operacje graficzne i tekstowe wykonywane były [redacted] oknie [redacted] numerze 0, czyli [redacted] całym ekranie monitora. Amos umożliwia jednak otwieranie



nowych okien o prawie dowolnych wymiarach. A oto komendy przeznaczone do sprawowania kontroli nad nowo otwieranymi oknami.

Wind Open a,x,y,w,h,r
a - Numer okna.
x,y - Współrzędne lewego górnego rogu okna.
w - Szerokość okna (podana w ilości znaków).
h - Wysokość okna (podana w ilości znaków).
r - Rodzaj ramki
Komenda ta otwiera nowe okno w podanym miejscu i o podanych wymiarach. Okno to staje się bieżącym dla wszystkich operacji graficznych i tekstowych. "r" jest numerem wzoru ramki wokół okna (od 0 do 15). Parametr ten można pominąć i tym samym otworzyć okno bez ramki.

Wind Save
Zabezpiecza przed usunięciem zawartość okna bieżącego na wypadek przykrycia go przez inne nowo otwarte.

Title Top a$
a$- Zmienna tekstowa zawierająca tekst do wyświetlenia na górnej krawędzi okna.
Wyświetlanie dowolnego tekstu [redacted] górnej krawędzi okna (na ramce). Działa tylko przy oknach z ramką.

Title Bottom [redacted]
a$ - Zmienna tekstowa
Działa analogicznie do Title Top umieszczając tekst na dolnej krawędzi okna.

Window a
a - Numer okna.
Ustala okno o numerze "a" jako bieżące (gdy mamy kilka otwartych okien).

Wind close
Usuwa okno bieżęce.

Clw
Kasuje zawartość okna i wypełnia je kolorem tła.
Możliwości wykorzystania opisanych komend dotyczących okien także zilustruje przykładem.

Okna z ramką (demo)

```
For R=1 To 16
Wind Open 1,10,10,20,20,R Cdown
Print "Ramka nr.";R
Cdown
Print "Wcisnij dowolny"
Print "klawisz !!!"
Title Top "Okno testowe"
Title Bottom "amiga"
Wait Key
Wind Close
```

Wszystkie przedstawione do tej pory komendy operujące na tekście, używały standardowego fontu zadeklarowanego [redacted] edytorze (Default.font). Nie oznacza to, że jesteśmy ograniczeni do jednej czcionki. Aby tego dokonać trzeba tylko użyć innych komend do wyświetlania tekstów. Na początek przedstawię instrukcje przygotowujące zestaw czcionek.

Get Fonts
Ładuje z dysku i pamięci ROM listę wszystkich dostępnych czcionek. Jest to konieczne przed ich wykorzystaniem.

FontsS(a)
a - Numer fontu.
Zmienna tekstowa zawierająca dane na temat fontu o numerze "a" z listy utworzonej komendą Get Fonts. Jeśli zestaw o podanym numerze nie istnieje zmienna jest ciągiem pustym. Zawartość tej zmiennej można wyświetlić np. tak:

Print Font$(1)
Dla fontu o numerze "1".

Set Font a
a - Numer fonfu.
Ustala czcionkę dla komendy "Text" i wczytuje ją z dysku. Wszystkie wyświetlane przez nią teksty bądą korzystały z fontu o numerze podanym przy Set Font.

Text x,y,a$
x,y - Współrzędne graficzne
a$ - Zmienna tekstowa zawierająca wyświetlany tekst.

Komenda [redacted] wyświetla tekst podany w zmiennej w punkcie o współrzędnych graficznych x,y. Należy pamiętać, że tekst tak wyświetlany jest traktowany jak grafika i do jego wyświetlania nie mają zastosowania komendy omówione na początku artykułu. Np. jego kolor ustala komenda Ink, a nie Pen i Paper, a w przypadku, gdy nie zmieści się w jednej linii, nie zostanie przeniesiony do następnej, jak to było poprzednio. "Nadwymiarowy" fragment zostanie po prostu ucięty i to w dowolnym miejscu. Myślę, że po kilku próbach nie będzie problemów z wykorzystaniem tej instrukcji. Jak zwykle na zakończenie będzie "pożyteczny" programik. Tym razem jest to "tester fontów" pokazujący wszystkie dostępne czcionki i drukujący testowy fragment w każdej z nich.

Tester czciouek

```
Hide
A=1
Screen Open 1,640,256,16,Hires
Paper 0
Clw
Centre "Zestaw dostępnych czcionek"
Locate 0,2
Get Fonts
Do
Print Font$(A)
If Font$(A)="" Then Exit
A=A+1
Loop Wind Open 2,0,200,79,7,2
Border ,0,
Curs Off
For B=1 To A-1
Locate 43,0
Print "CTRL+C * STOP *"
Locate 43,1
Print "Lewy klawisz myszki * NASTĘPNY *"
Home
Pen 3
Print Font$(B)
Pen 2
Set Font B Ink 2,0
Text 30,240,"Commodore AMIGA"
Do
Wait 3
While Mouse Key=1
Exit 2
Wend
Loop
Clw
Next B
```

[redacted] będę dokładnie opisywał jego działania, aby zachęcić wszystkich do wpisania i uruchomienia tego programu. Jak zwykle dokonywanie wszelkich modyfikacji jest dozwolone, a nawet wskazane. Życzę powodzenia i namawiam do czytania następnego odcinka, poświęconego współpracy programów pisanych w amosie ze stacją dysków.

Chciałbym wrócić jeszcze [redacted] chwilę do tematu tworzenia stref aktywnych, poruszanego w drugim odcinku naszego kursu. Konkretnie, ustalania parametrów dla instrukcji Set Zone. Zasugerowana przeze mnie analogia do parametrów instrukcji Box, rysującej prostokąty jest jak najbardziej słuszna, ale jak trafnie zauważył Pan Tadeusz Figiński z Gdańska w niedawno nadesłanym do redakcji liście, jej zastosowanie w praktyce może okazać się dosyć pracochłonne. Zaproponował zatem zastosowanie programu pomocniczego, podającego parametry położenia wskaźnika myszki we współrzędnych ekranowych. Oto on:

```
Do
X=X Screen(X Mouse)
Y=Y Screen (Y Mouse)
Locate 0,0 : Print X;"  "
Locate 0,1 : Print Y;"  "
If Mouse Key Then Edit
Loop
```

Po wczytaniu przygotowanej grafiki i uruchomieniu podanego wyżej programiku, wystarczy wskazywać potrzebne nam punkty i zapisywać pojawiające się w lewym górnym rogu ekranu współrzędne. Mając zapisane wszystkie potrzebne liczby, można już bez przeszkód przystąpić do deklarowania stref we własnym programie. Chcę jeszcze serdecznie podziękować Panu Tadeuszowi za list i wyrazone w nim uznanie dla naszego pisma. ❑

Kurs Arexxa

część 1

Izabela Skibińska

Równocześnie [redacted] pojawieniem się systemu operacyjnego 2.0 firma Commodore dostarcza interpreter Arexxa na swoich dyskietkach systemowych. Już sam ten fakt nasuwa, słuszne zresztą, przypuszczenie o dużej użyteczności tego programu w systemie operacyjnym Amigi.

Arexx należy do pascalopodobnej grupy języków programowania. Jego składnia jest nieskomplikowana, [redacted] większość instrukcji analogiczna do tych, jakie spotykamy w C czy Pascalu.

Podstawowa różnica polega na tym, [redacted] jest to język interpretowany, czyli [redacted] wykonania instrukcji programu niezbędna jest obecność w systemie procesu interpretera (Rexxmast). Jakie więc korzyści może nieść w sobie ten kolejny język wysokiego poziomu wolniejszy od innych ze względu na konieczność interpretacji? Otóż w wielozadaniowym systemie operacyjnym Amigi, Arexx jest wspaniałym rozwiązaniem na komunikację między różnymi procesami. Takie udogodnienia jak sterowanie programami "od zewnątrz", automatyczne uaktualnianie danych, są niemalże w zasięgu ręki!

Dzięki zaopatrzeniu niektórych programów w interface Arexxa, możemy uniknąć wielu godzin żmudnej pracy. O przykłady nietrudno. Wyobraźmy sobie, że chcemy zrobić zestaw polskich fontów PostScriptowych pod programem TypeSmith. Prosty skrypt w Arexxie umożliwi nam wyrzucenie niepotrzebnych znaków z miejsc gdzie mają być polskie literki i skopiowanie tam znaków, do których też przy pomocy Arexxa dokopiujemy odpowiednie części. Stanowi to 80% całej pracy i do tego jej najżmudniejszej części.

Naturalnie w całej tej komunikacji międzyprocesowej uzyskiwanej przez użycie Arexxa wykorzystywany jest cały mechanizm message portów i messages dostarczany przez exec. library i znany wszystkim zajmującym się choć trochę programowaniem systemowym. Z grubsza dla niezorientowanych postaramy, się przedstawić ideę komunikacji procesów [redacted] Amidze.

Cała wymiana informacji polega [redacted] tym, że do portu związanego z danym procesem (port ten jest mu przypisany przez system w momencie powstania) mogą "przychodzić" informacje (messages) od innych procesów. Użyte tu słowo oznacza dokładnie to, że procesowi otrzymującemu message zostaje udostępniony jej adres. Ma on teraz całkowitą kontrolę nad pamięcią okupowaną przez message. Ponieważ pamięć [redacted] była zaalokowana przez inne zadanie, pozostaje jeszcze zwrócić właścicielowi kontrolę nad tą pamięcią. Oczywiście wszystkim zajmuje się system, a do programisty należy odpowiednie wywołanie funkcji systemowych. Wracając do Arexxa, to proces interpretera posiada port identyfikowany przez nazwę "REXX".

Zgodnie z tym, co napisałam wyżej o komunikacji między procesami, każdy program, który "rozumie" Arexx używa dodatkowego portu o unikalnej nazwie, będącej jego identyfikatorem. Podczas wykonywania skryptu Arexxa, między portami interpretera i programu, zachodzi odpowiedni "dialog" za pośrednictwem specjalnej message, wzbogaconej w stosunku do zwykłej struktury message o dodatkowe informacje pochodzące z exec.library. Jednak to od programu zależy jakie rozkazy akceptuje i jak [redacted] brzmią. Inne rozkazy, których program nie rozumie wysyłane są do portu "REXX", czyli właściwego interpretera Arexxa. Jest to oczywiście podejście bardzo uproszczone. Dokładniej tym problemem zajmiemy się w dalszej części kursu przy omówieniu zasad tworzenia własnych programów z interfacem Arexxa.

Do kolejnych zalet Arexxa, związanych z bardzo dobrym wykorzystaniem możliwości systemu operacyjnego Amigi, należy jego organizacja. Interpreter korzysta bowiem z biblioteki typu *run-time-shared* - *rexxsyslib.library*, [redacted] umożliwia jednoczesne (bezkonfliktowe) wykonywanie dowolnej liczby programów Arexxa. Tak naprawdę, to w tej właśnie bibliotece znajdują się główne funkcje całego interpretera.

Do wykonania jakichkolwiek skryptów Arrexa niezbędne jest uruchomienie programu RexxMast. Zadaniem tego programu jest przygotowanie środowiska, a więc umieszczenie *rexxsyslib.library* na liście bibliotek systemowych. Poza tym Arexx będąc, de facto, [redacted] systemowej liście bibliotek, staje się integralną częścią systemu operacyjnego Amigi i nie ma potrzeby każdorazowego uruchamiania interpretera, jak to się dzieje np. w przypadku AmigaBasic. Niezwykle pożyteczną sprawą jest możliwość dodawania tego typu bibliotek do środowiska Arexxa. Mogą być tam zaimplementowane funkcje systemowe, a dzięki temu Arexx oprócz swoich "wrodzonych" umiejętności, (skrypty i komunikacja między procesami) może być kolejnym językiem programowania, w którym bez problemu możemy tworzyć własne programy użytkowe.

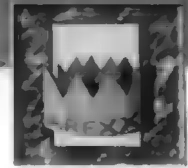
Sam fakt, że Arexx jest interpreterem niesie, wbrew pozorom, również wiele zalet. Przy olbrzymiej ilości prostych skryptów, każdorazowa kompilacja i konsolidacja byłaby czystą nieoszczędnością miejsca i czasu. Poza tym wykonywanie programu krok po kroku w znaczny sposób ułatwia proces jego "odpluskwiania". Temu zagadnieniu również poświęcę sporo miejsca.

Żeby z powodzeniem wykonywać Arexxowe skrypty niezbędna jest odpowiednia instalacja systemu. Na niektórych modelach Amig (np. A3000) program ten już jest zainstalowany. Posiadacze innych modeli muszą wykonać parę nieskomplikowanych czynności:

1. Stworzyć katalog o dowolnej nazwie (np. makedir dh0: Rexx).
2. Skopiować tam wszystkie pliki dostarczane [redacted] Arexx-em (HI, RX, RXC, RXLIB, RXSET, TCC, TCO, TE, TS, WaitForPort - opis patrz tabelka) i oczywiście [redacted] interpreter - *Rexxmast*
3. Skopiować do katalogu *libs:* biblioteki *rexxsyslib.library* i *rexxsupport.library* W przypadku, gdy w katalogu libs: nie [redacted] mathieeedoubbas.library należy ją tam koniecznie dograć.
4. Stworzyć przypisanie do katalogu Rexx - assign REXX: dh0:Rexx
5. Rozkazem path *Rexx:* add dołączamy katalog *Rexx:* do listy katalogów przeglądanych przez system w poszukiwaniu wprowadzonych przez użytkownika rozkazów.
6. Pozostaje już tylko uruchomienie właściwego programu RexxMast

Oczywiście czynności opisane w punktach 4, 5 i 6 najlepiej wpisać do startup-sequence. O korzyściach chyba nie należy wspominać.

Możemy teraz wywołać dowolny program Arexxa rozkazem rx nazwa programu. Do nazwy programu Arexxowego możemy dodać rozszerzenie ".rexx". [redacted] to jednak znaczenie czysto informacyjne. Przy wywoływaniu programu możemy je pominąć, ale tylko pod takim warunkiem, że brzmi ono rzeczywiście "rexx". Gdy Polecenie rx nie znajdzie programu w katalogu aktualnym, kontynuuje poszukiwania w *REXX:*. Taki jest [redacted] istnienia tego katalogu w systemie. Oczywiście nie trzeba umieszczać plików systemowych ArexxA tylko i wyłącznie w *REXX:*. Równie dobrze mogą być one w katalogu [redacted] dowolnej nazwie, wystarczy uwzględnić fakt przy dodawaniu ścieżki do systemu (patrz punkt 5).



Jak zwykle w kursach języków programowania, rozpoczniemy od wypisywania i wczytywania zmiennych, a także poznamy składnię podstawowych instrukcji.

Przedtem jednak parę słów o konwencji zapisu skryptów Arexxowych. Każdy program w Arexxie musi zaczynać się komentarzem, czyli dowolnym ciągiem znaków (zerowym też) zawartym między dwoma sekwencjami znaków: "/*" i "*/". Komentarze mogą być zagnieżdżone. Rozkazy Arexxa mogą być pisane zarówno dużymi jak i małymi literami. Instrukcje możemy pisać linia pod linią oraz obok siebie odczielone średnikiem.

```
/* Przyład 1 */

parse arg licznik
say 'przykład1"
say 'argument :'licznik
ay "wprowadź liczbę"
pull liczba

do i = 1 to licznik+liczba
   say 'licznik : "i
end
liczba = 'string1"
licznik = "string2'
ay "    "liczba || licznik"      'licznik
```

Argumenty podawane przy wywołaniu skryptu Arexxowego mogą być przejęte dzięki dwóm rozkazom parse i arg. Po nich następuje lista zmiennych, pod które mają być podstawione argumenty. W naszym przypadku obojętnie jaki argument wpiszemy, po wywołaniu wyżej przedstawionego skryptu, będzie podstawiony pod zmienną licznik.

Arexx jest w odróznieniu od Pascala językiem wybitnie liberalnym jeżeli chodzi o traktowanie zmiennych. Po pierwsze nie musimy deklarować zmiennych. Po drugie są one beztypowe. Zmienna powstaje w momencie odwołania się do niej. Niezainicjalizowana, przyjmuje swoją nazwę za zawartość. Rozkaz say wyprowadza swoje argumenty na wyjście. Jak widać z ostatniej linii naszego przykładu, mogą być to przeplatane łańcuchy znaków i zmienne. Rozkaz PULL będący skrótem rozkazu PARSE UPPER PULL (dokładą analizą rozkazu PARSE wkrótce się zajmiemy na łamach naszego kursu) wczytuje znaki z wejścia. Słowo kluczowe DO pełni w Arexxie podwójną rolę. Przede wszystkim rozpoczyna ciąg instrukcji, który kończymy słowem kluczowym END. Można porównać ją z pascalowską instrukcją grupującą begin-end. Umożliwia także tworzenie pętli. Pełna definicja wygląda następująco:
DO [[Var = i]|[ZM] [TO i] [BY i]] [FOR i][FOREVER][WHILEcond|UNTIL cond].

Jak widać wszystko to, co następuje po instrukcji DO jest opcjonalne (nawiasy kwadratowe). Samotne słowo kluczowe DO spowoduje wykonanie ciągu instrukcji po nim następującego do instrukcji END dokładnie jeden raz. Parametry po DO określają warunki dla iteracji.

"Var = i" - przypisuje zmiennej licznikowej wartość początkową. Przypisanie to jest konieczne gdy korzystamy z wariantu pętli zawartego w pierwszym nawiasie kwadratowym. Oznacza to, że słowa kluczowe TO, po którym następuje ograniczenie zmiennej licznikowej i BY - krok zmiennej licznikowej, nie mogą występować samodzielnie. Zamiast tej konstrukcji możemy podać tylko wyrażenie ZM. Efekt podobny jak przy użyciu FOR. Oczywiście po tym wariancie pętli mogą występować inne warunki narzucane na pętle (z wyjątkiem FOREVER). Stwarza to możliwość elganckiego zapisu rozbudowanych warunków. (Patrz przykład.)
FOR i - wyrażenie i określa liczbę powtórzeń pętli.
FOREVER - pętla wykonuje się w nieskończoność. Możliwość wyjścia poprzez BREAK lub LEAVE.
WHILE cond - wartość logiczna wyrażenia cond obliczana jest na początku każdej iteracji. Gdy wynik wynosi 1 - kontynuacja, 0-zatrzymanie
UNTIL cond - wartość logiczna wyrażenia cond obliczana jest na końcu każdej iteracji. Gdy wynik wynosi 0 - kontynuacja, 1-zatrzymanie

```
            /* Przykład 2*/

   /* Szukanie najmniejszego dzielnika danej liczby */
   /* Z ograniczeniem prób do dziesięciu */
say "Wprowadź liczbę"
pull n
do i=2 to n for 10 until n//i=0
say i
end
```

Na poparcie pochlebstw pod adresem Arexxa, a szczególnie jego doniosłej roli w zakresie sterowania programami, przygotwaliśmy na zakończenie, skrypt, sterujący programem ADPRO. Skrypt ten zamienia dowolną ilość obrazków w formacie IFF (wybór z shiftem) na odcienie szarości. *UWAGA! Zapamiętuje pod tą samą nazwą.*

Na razie wiele rzeczy może wydać się nie jasne. Nic dziwnego, aby pisać skrypty w Arexxie, sterujące programami posiadającymi port Arexxa, oprócz samej znajomości tego języka, niezbędny jest opis ze stawu rozkazów jakie akceptuje dany program. Sterowaniu programem ADPRO poświęcimy jeden z najbliższych odcinków naszego kursu o Arexxie.

```
/* F1.adpro */
/* Skrypt umożliwiający dokonywanie operacji zamiany dowolnej liczby */
/* obrazków formatu IFF na odcienie szarości */
/* Uruchomiene spod ADPRO przez klawisz F1 - patrz nazwa programu ! */
LFORMAT "IFF"
SFORMAT "IFF"
mmmADDRESS "ADPro"
OPTIONS RESULTS

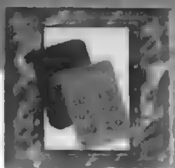
GETFILES
IF RC ~= 0 THEN DO
EXIT
END

LIST = ADPRO_RESULT
COUNT = WORDS(ADPRO_RESULT)
I=1

DO WHILE (I <= COUNT)
PLIK = WORD(LIST,I)
LOAD PLIK
OPERATOR 'COLOR_TO_GRAY"
EXECUTE
OKAY1 PLIK
SAVE PLIK "IMAGE"
I=I+1
END
```

Tabelka: Programy związane ze środowiskiem Arexxa.

REXXMAST	**instaluje Arexxa w systemie**
RX (nazwa pliku) (argumenty)	**wysyła skrypt Arexxowy do interpretera wraz z argumentami**
HI	**zarzymuje wszystkie wykonujące się operacje Arexxa**
RXC	**wyrzuca z systemu rezydentny proces interpretera (Rexxmast i rexxsyslib.library)**
TCO	**przekierowuje wyjście z procesu śledzenia programu do Global-Tracing-Console**
TCC	**zamyka Global-Tracing-Consle**
TS	**proces śledzenia będzie odbywał się w trybie interakcyjnym**
WaitFor Port name	**czeka 10 sekund na "oznaki życia" portu o nazwie name. Zwraca 0 gdy port istnieje, 5 w p.p.**
REXXSET (NAME ((=) VALUE))	**wprowadza do Clip Listy zmienną NAME o wartości VALUE. W przypadku nie podania wartości zmienna jest usuwana**
TE	**wyłącza znaczniki śledzenia we wszystkich aktualnie wykonywanych programach Arexxowych**



PRAKTYCZNE NARZĘDZIA

Maciej Klimkiewicz

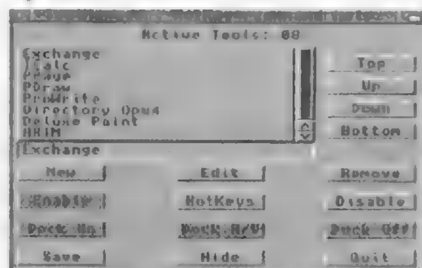
Kiedy w 1946 roku powstała pierwsza elektroniczna maszyna cyfrowa o nazwie ENIAC, podekscytowanie naukowców tym wynalazkiem wydawało się zupełnie nieuzasadnione. Dla zwykłego zjadacza chleba plątanina kabli i lamp elektronowych była czymś obcym. Jednak, gdy po kilkunastu latach przed klawiaturami zasiedli pierwsi komputerowi amatorzy, stało się jasne, że to właśnie oni będą dyktować kierunek rozwoju tej dziedziny.

Dzięki nim główny nacisk postawiono na uproszczenie obsługi komputera. Spowodowało to prężny rozwój programów narzędziowych (ang. tools), który trwa do dnia dzisiejszego.

Programy narzędziowe mają za zadanie uprościć obsługę systemu komputerowego albo rozszerzyć jego możliwości. Jednym z nich jest ToolManager.

Służy on do tworzenia menu użytkownika i specjalnych ikon "dotykowych" umożliwiających szybkie uruchamianie programów bez konieczności skakania po katalogach. Na Rysunku 1 jest przedstawione główne okno służące do definiowania nowych obiektów (New) lub ich wymazywania (Remove). Każdą pozycję menu można dowolnie przesuwać do góry (Up) albo na dół (Down) bądź na początek i koniec listy (Top, Bottom).

Rysunek 1



Gadżetem Edit przechodzi się do okna szczegółowo definiującego obiekt (Rysunek 2). Na początku należy wpisać do (*Real Name)* jaki program ma być uruchomiony i w jakim katalogu roboczym ma pracować (*Work Dir*). Natomiast w (*Alias*) umieszcza się nazwę pod jaką ma widnieć w spisie. Nie jest wymagane, żeby nazwa pokrywała się z tytułem wywoływanego programu.

Poza tym trzeba jeszcze określić w jaki sposób ma być zasymulowane uruchomienie programu (*Type*). Są dwie możliwości: tak jak kliknięciem w Workbenchu na ikonę (trzeba wtedy uaktywnić Wb), albo wydaniem komendy z *Shella* - (CLI). Jest to bardzo przydatna funkcja, ponieważ niektóre programy muszą być wywoływane w konkretnie określony sposób.

Istnieje również możliwość zdefiniowania ścieżki dostępu dla programu (Path) oraz pliku wyjściowego (OutFile). Jednak te opcje są aktywne tylko dla programu startującego z Shella.

Jeżeli użytkownikowi zależy na nadaniu własnej wielkości stosu programu, może wpisać rządaną wartość do (Stack).

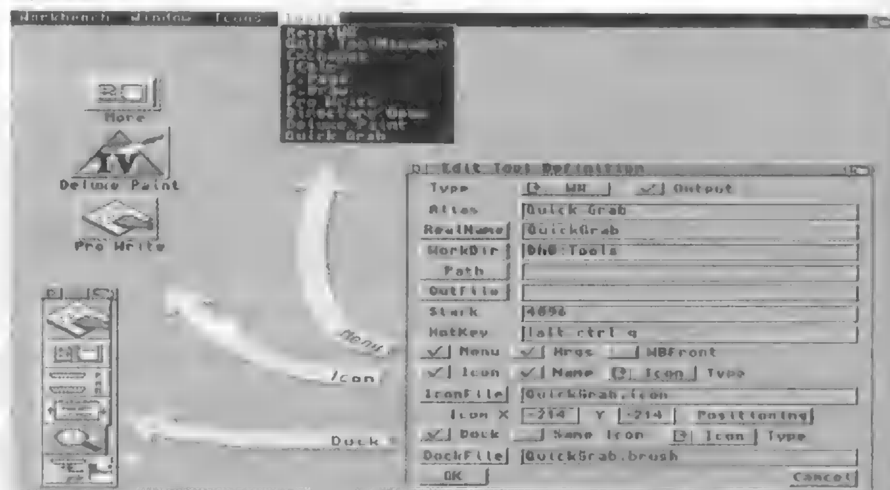
Żeby lepiej zrozumieć o co tu chodzi, prześledźmy pewien przykład. Załóżmy, że chcemy wywoływać następujące polecenie:

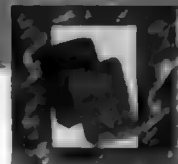
c:list >ram:spis dh0:

Należy wtedy ustawić typ wywołania jako (Type - CLI) i wpisać do odpowiednich wierszy następujące wyrażenia:

(RealName) - c:list, (WorkDir) - dh0:, (OutFile) - ram:spis, a do (Alias) do

Rysunek 2





wolną nazwę pod jaką to polecenie ma występować w menu.

To samo uruchomienie można uzyskać wieloma innymi sposobami, np:

(RealName) - c:list dh0:
(OutFile) - ram:spis

albo jeszcze inaczej:

(RealName) - c:list >RAM: dh0:

Jak doskonale widać program daje użytkownikowi duże możliwości do popisu. Wszystkie opisane do tej pory funkcje służyły do określenia sposobu uruchomienia programu od strony systemowej. Jednak Amigowca najbardziej interesuje w jaki sposób będzie mógł wydać polecenie jego "odpalenia". Jak już zostało na wstępie napisane, możliwe to jest z menu (*Menu*), okna reagującego na dotyk (*Dock*), wykładanej ikony (*Icon*) i przez naciśnięcie kombinacji klawiszy (*HotKey*). Wszystkie te sposoby można mieszać ze sobą.

Ciekawą możliwością jest to, że przy włączonej opcji (*Menu*) i (*Args*) program może być wywoływany z argumentami. Wystarczy wtedy zaznaczyć wybraną ikonę i wybrać z menu zdefiniowaną pozycję. Jeżeli np. zostało wpisane:

(Alias) - Listowanie, (RealName) - c:list, (Menu) i (Args) - są uaktywnione,

to po zaznaczeniu jakiejś ikonki - np: "Ram Disk" i wybraniu z menu pozycji "Listowanie" na ekranie ukaże się spis RamDysku. Radzę poeksperymentować.

Drugim sposobem na uruchamianie programów są ikony, dla których istnieją dwie możliwości - tradycyjnie tak jak przez "*Leave Out*" (*Icon*), albo przy pomocy nowego, specjalnego okna, w którym do uruchomienia wystarczy samo dotknięcie obrazka programu znajdującego się wewnątrz osobnego okna. Oba sposoby umożliwiają "podłożenie" pod program gotowej ikony (*Type Icon*) albo zwykłego pędzla (*Type Brush*). Dzięki temu rzeczywista ikona programu nie musi zgadzać się z "podłożoną". Daje to możliwość np. zminiaturyzowania wyłożonych ikon, pomimo że w katalogach będą miały zwykłe rozmiary.

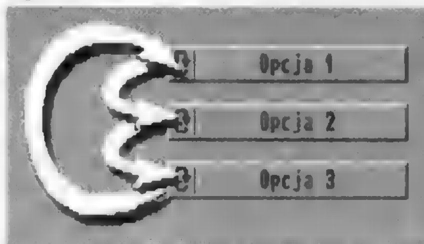
Pomimo, że ToolManager został napisany w 1991 roku, swoimi możliwościami przewyższa wiele nowszych tego typu narzędzi. Żadko zdarza się, żeby narzędzie obsługujące menu oferowało tak wiele różnych możliwości. Moim zdaniem wadą ToolManagera jest zbyt mała ergonomiczność, która sprawia że programowanie w nim obiektów - zwłaszcza na początku, wydaje się trochę zawiłe.

Kolejnym programem narzędziowym godnym uwagi jest CycleToMenu.



Jest to typowe małe narzędzie, które nawet nie otwiera własnego okna podczas uruchamiania, dlatego też wszelka komunikacja pomiędzy nim, a użytkownikiem musi odbywać się przez Exchange'a. Jedyną czynnością jaką wykonuje CycleTo Menu jest rozszerzenie możliwości tzw. przewijanych gadżetów. Normalny przewijany gadżet wygląda jak na rysunku 3.

Rysunek 3



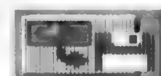
Klikając w niego wskaźnikiem myszki wywołuje się po kolei wszystkie zaprogramowane w nim opcje. Po dojściu do końca listy automatycznie następuje przeskok do pierwszej pozycji. W momencie, kiedy ilość opcji w gadżecie jest dość duża, wyszukanie i ustawienie wybranej może być czasochłonne. W tym momencie z pomocą przychodzi CycleToMenu, który po uruchomieniu podmienia część programu z systemu odpowiedzialnego za obsługę przewijanych gadżetów na swój własny kod. Od tego momentu kliknięcie i przytrzymanie wskaźnika na gadżecie powoduje rozwinięcie się menu ze spisem dostępnych opcji (Rysunek 4).

Rysunek 4



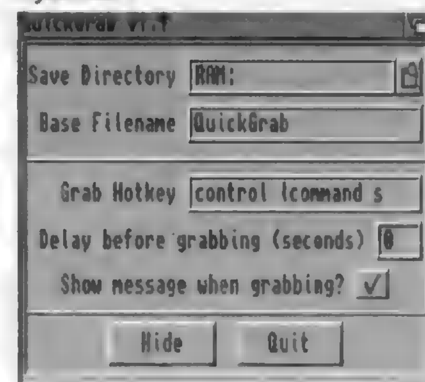
Trzymając w dalszym ciągu lewy klawisz myszki można bezpośrednio najechać i uaktywnić wybraną opcję.

Na zakończenie chciałbym przedstawić jeszcze jeden program narzędziowy o nazwie QuickGrab.



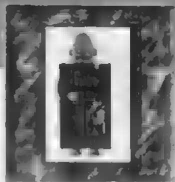
Służy on do wyciągania obrazków z ekranu (Rysunek 5). Odbywa się to w ten sposób, że po wciśnięciu odpowiedniej kombinacji klawiszy zdefiniowanej w (Grab Hotkey), aktualnie widoczny ekran jest nagrywany do wskazanego katalogu określonego w (Save Directory). Wszystkie stworzone w ten sposób obrazki mają nazwę zaczynającą się od słowa zapisanego w (Base Filename) z końcówką określającą numer kolejny obrazka, np: QuickGrab.000, QuickGrab.001 itd. Bardzo przydatną funkcją oferowaną przez to narzędzie jest możliwość zdefiniowania czasu, jaki ma upłynąć od momentu wciśnięcia klawiszy do rozpoczęcia nagrywania obrazka. Dzięki temu można w bardzo prosty sposób nagrywać np. pull-down menu danego ekranu. Dziwne, że tak niewiele programów tego typu oferuje tę przydatną funkcję.

Rysunek 5



Główną wadą QuickGrabera w stosunku do jego konkurencji jest brak opcji wycinania wybranego fragmentu ekranu. Jednak prostota obsługi, oraz możliwość definiowania opóźnienia w czasie, powodują że QuickGraber dobrze spełnia swoje zadanie. Najlepszym na to przykładem jest ten artykuł, w którym wszystkie grafiki zostały wyciągnięte przy jego pomocy.

I to by było na tyle. W następnym odcinku kolejna porcja programów narzędziowych obsługujących menu, drukarkę i nie tylko. Do zobaczenia za miesiąc. ❑



Separacje barwne

Krzysztof [illegible]

Wraz z rozpowszechnieniem się programów typu Desktop Publishing coraz więcej osób próbuje na własną rękę wykonywać różnego rodzaju gazetki czy reklamówki. Nierzadko w zamyśle twórcy jest to, aby praca w końcowym wydruku była kolorowa. Najczęściej jednak wszystko wygląda pięknie, dopóki dzieło pozostaje na ekranie komputera. Gdy chcemy uzyskać efekt na papierze, pojawiają się problemy.

Przede wszystkim drukowanie na kolorowych drukarkach igłowych, czy atramentowych odpada, gdyż jakość oferowana przez te urządzenia jest czysto domowa. Jeśli zaś wziąć pod uwagę kolorowe drukarki laserowe i termiczne, to co prawda jakości nic nie można zarzucić, lecz przy ilości kopi np. 500 egzemplarzy cena zbija z nóg. Pozostaje nam więc druk na maszynie offsetowej.

Przygotowanie do takiego druku pole[illegible] na wykonaniu klisz służących jako podstawa do uzyskania płyt drukarskich. Klisze uzyskujemy dzięki podłączeniu komputera do fotonaświetlarki. Ponieważ urządzenie to do tanich nie należy, najczęściej korzysta się z niego dzięki uprzejmości firm posiadających i oczywiście za odpowiednią opłatą.

Załóżmy, że uporaliśmy się już z problemem dostępu do fotonaświetlarki i jesteśmy dogadani z drukarnią. Należy zastanowić się jeszcze jaki program wybrać do naszych celów? To znaczy z jakiego uzyskamy jak najlepsze reprodukcje barwne? Ocenie poddamy dwie najbardziej popularne aplikacje DTP, czyli: Professional Page 4.1 i Page Stream 2.22. Lecz najpierw trochę teorii:

Jak jest drukowany kolor?

Na ekranie kolory reprezentowane są przez trzy składowe barwy: czerwoną (Red), zieloną (Green) i niebieską (Blue). Jest to model addytywny (RGB), gdyż maksymalne nasycenie każdej z nich daje kolor biały (kolory świecą). Na papierze natomiast występuje subtraktywny model barwny ze składowymi: błękitną (Cyan), purpurową (Magenta) i żółtą (Yellow). Gdy każda z nich przyjmie wartość 0, otrzymamy kolor biały (papieru). Dodatkowo w druku stosuje się jeszcze kolor czarny (Kontur), gdyż - mimo że teoretycznie maksymalne nasycenie pozostałych daje również barwę czarną to jednak niedoskonałość farb oraz wchłanianie farby przez papier powoduje, że występują trudności z uzyskaniem czystego czarnego. Drugim powodem stosowania czerni, jest druk tekstu. Korzystanie z kombinacji trzech kolorów w celu jego uzyskania znacznie zwiększyłoby koszty produkcji.

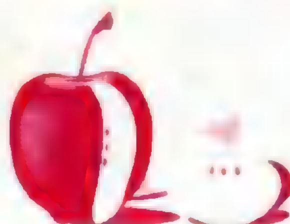
Z powyższego rozumowania wynika, że każda kolorowa ilustracja w dokumencie musi zostać przez program rozdzielona na cztery warstwy kolorystyczne CMYK. Operacja ta nosi nazwę separacji barwnej (Rysunek 1). Każda z warstw posiada odcienie np. odcienie błękitu (Cyanu), od białego do pełnego nasycenia. Jak to można uzyskać, przecież farba jest jedna i na dodatek nakładana z takim samym naciskiem przez maszynę drukującą? Otóż w tym przypadku stosuje się tak zwane rastrowanie, czyli tworzenie siateczki punktów. Wielkość tych punktów decyduje o nasyceniu koloru, punkty ułożone są na niewidocznych liniach (Rysunek 2). Odległość między liniami rastra określa się w ilości lini występujących w calu (lpi) lub w centymetrze (l/cm) i nosi nazwę "gęstość lini rastra". Wielkość [illegible] odpowiada za największy wymiar punktu w rastrze. Po nałożeniu na siebie wszystkich czterech zrastrowanych warstw okaże się, że kolejnym ważnym elementem jest kąt pod jakim linie rastra są ułożone (Rysunek 3). Tradycyjne kąty ustawiane dla poszczególnych kolorów wynoszą: 105° Cyan, 75° Magenta, 90° Yellow i 45° Kontur (czarny). Obydwa opisywane programy mają możliwość ustawienia kątów, jak i gęstości rastra. Professional Page nie nastręcza problemów, gdyż wszystkie parametry mamy podane jak na dłoni. Natomiast twórcy programu Page Stream nie popisali się. W Page Streamie należy mianowicie w oknie dialogowym "Configure Printer" wpisać właściwą sekwencję z podanymi kątami i gęstościami. Na dodatek schemat wpisywania jest ściśle określony, dzięki czemu bez rozłożonej obok komputera instrukcji obsługi programu, nie ma co zabierać się za ustawianie parametrów. Jest również plus takiego podejścia do sprawy, ponieważ istnieje możliwość wpisania poleceń tak, aby nie wszystkie strony dokumentu podlegały rozbiciu (np. pierwsze trzy, dwie bez rozbić i następne pięć znowu z separacją). Takiej możliwości Pro Page nie posiada. Nie

Program	Wielkość dokumentu	Czas druku	Czas odświeżania ekranu przy zmianie powiększenia	Zajętość RAM	Wielkość pliku postscriptowego
Professional Page	141kB	18,5 min.	9 sek.	847kB	5,58MB
Page Stream	1959kB	20 min.	18 sek.	2054kB	5,52MB

Rysunek 1.
Na ilustracjach poniżej widać poszczególne warstwy rozseparowanego obrazka jabłka.



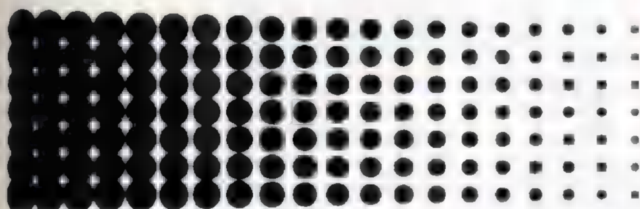




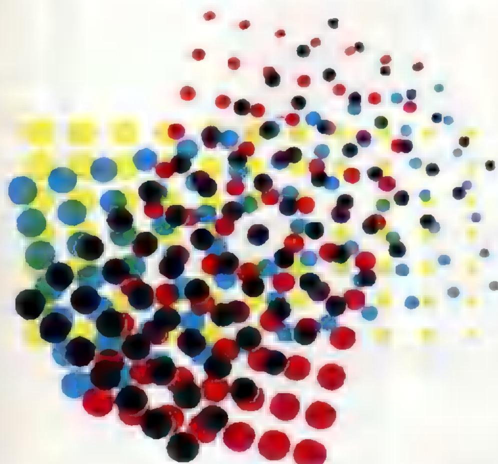




Rysunek 2.
Punkty rastra są poukładane na równo oddalonych od siebie liniach.



Rysunek 3.
Dzięki różnym kątom rastra unikamy niepotrzebnego nakładania się punktów na siebie - tworzy się wzorek.

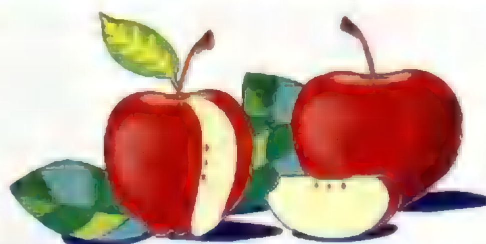




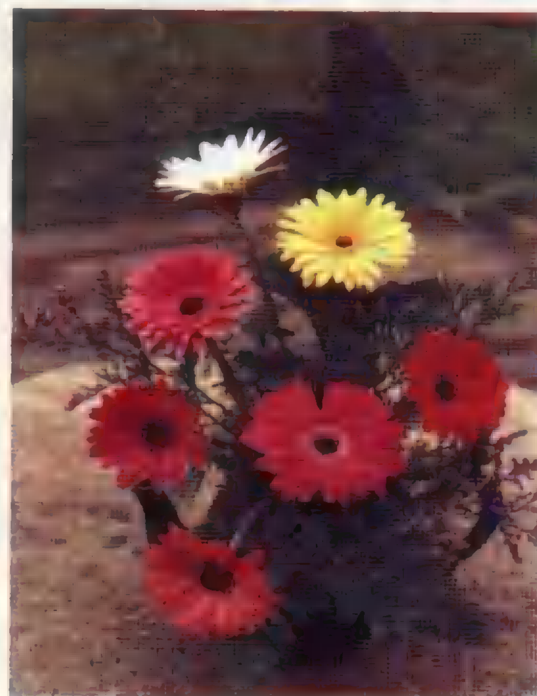
Page Stream 2.22







Professional Page 4.1

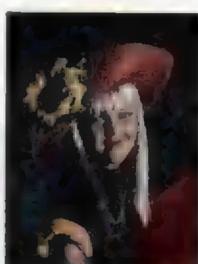








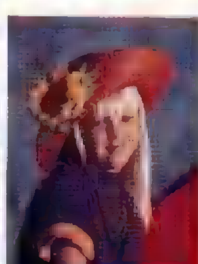
UCR=0 GCR=100



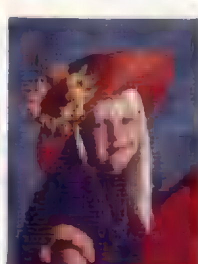
UCR=0, GCR=200



UCR=100, GCR=100



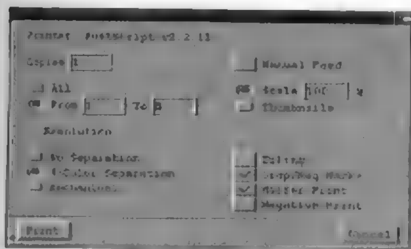
UCR=100, GCR=200



UCR=75, GCR=50

Powyżej zostały przedstawione wydruki z separacji wykonanych za pomocą programów Page Stream i Pro Page. Rozbicia z Pro Page dokonano z ustawionym UCR na 75, a GCR na 125. Obok natomiast separacje z Pro Page z różnymi ustawieniami UCR/GCR.





mniej takie ustawianie parametrów jest bardzo niewygodne i chyba jedynym wytłumaczeniem takiego zaimplementowania tej funkcji, jest chęć zabezpieczenia się przed piratami.

Ważnym parametrem jest także ustawiona rozdzielczość fotonaświetlarki. Decyduje ona o rozmiarze najmniejszego punktu w rastrze. Ani Pro Page, ani Page Stream nie mają możliwości zmieniania tej rozdzielczości, lecz na szczęście można ją ustawić w samej naświetlarce (Page Stream 3.0 będzie posiadał taką możliwość). Możemy zatem kontrolować wielkość punktów (największą poprzez gęstość rastra i najmniejszą przez ustawienie naświetlarki). Parametry te powinny być dopasowane do papieru na jakim będzie nasza praca drukowana. Im gorszy papier tym mniejsza musi być ustawiona rozdzielczość naświetlarki oraz gęstość rastra. Przykładowo dla papieru gazetowego nie ma [redacted] ustawiać naświetlarki [redacted] 2400 dpi, gdyż wszystkie małe punkty zniknął [redacted] chropowatej powierzchni. Podobnie gęstość rastra 150 lpi spowoduje, że cała ilustracja rozaleje się.

Zależność pomiędzy rozdzielczością [redacted] gęstością rastra decyduje o ilości uzyskanych odcieni danego koloru np. naświetlarka 1693 dpi, gęstość rastra 120 lpi i otrzymujemy 200 odcieni (maksymalnie stosuje się 256 stopni). Ponieważ występują związki między rozdzielczością, kątami i gęstością rastra, wielu producentów naświetlarek zaleca dokładne parametry. Ze swojej strony podaję kilka przykładowych ustawień dla trzech popularnych naświetlarek: Linotronic 300, Linotronic L 330R i Compugraphic CG 9800.

Jak już powiedzieliśmy wcześniej, na ekranie kolory świecą, a w wydruku widzimy je dzięki światłu odbitemu od powierzchni zadrukowanej kartki. W związku z tym, ten sam kolor widziany na monitorze i ten sam na papierze wydaje się nam diametralnie różny. Aby poradzić sobie z tym problemem należy dokonać kalibracji obrazu ekranowego czyli maksymalnego upodobnienia koloru ekranowego do wydrukowanego na papierze. Niestety, żaden [redacted] programów DTP na Amigę nie pomaga nam w tej operacji i jeżeli nie posiadamy monitora z możliwością kalibracji kolorów, to nic na to nie poradzimy. Na szczęście przy wykorzystywaniu grafiki bitowej w dokumencie sparwa ma się lepiej, gdyż oba programy separują ją z pewnymi korekcjami, które teoretycznie mają za zadanie zachować wierność barw. Oprócz tego podczas rozbić ilustracji bitowych często zachodzi potrzeba ingerencji w ilość koloru czarnego. Inaczej mówiąc chodzi o to, czy kolor czarny w druku ma powstawać z kombinacji trzech barw podstawowych CMY, czy z samej farby czarnej K, czy też w kombinacji wszystkich CMYK z ustawianą proporcją ich występowania. Korekcje odpowiedzialne za tę funkcję, noszą nazwę Under Color Removal (UCR) i Gray Component Replacement (GCR), a możliwość zmiany parametrów tych parmetrów występuje tylko w programie Professional Page.

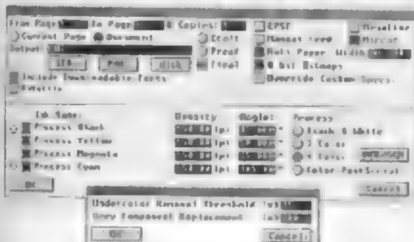
Działanie UCR polega na odjęciu od każdej barwy składowej pewnego procentu nasycenia w zależności od najmniejszej z wartości. Załóżmy, że mamy kolor składajacy się z 50% Yellow, 40% Magenta, 20% Cyan, 26% Czarnej. Jeśli UCR ustawimy na 50% to zostanie odjęta połowa najmniejszej wartości, czyli 10%. Otrzymamy więc 40% Yellow, 30% Magenta, 10% Cyan, 13% Czarnej.

Parametr GCR służy do procentowego przekazania częci wartości barw składowych CMY do barwy czarnej K. Weźmy tę samą barwę co poprzednio. Jeśli GCR ustawimy na 200% to otrzymamy 50% Yellow, 40% Magenta, 20% Cyan, 52% Czarnej. Przy GCR 50% będzie już 50% Yellow, 40% Magenta, 20% Cyan, 13% Czarnej.

Test

Do prób druku barwnego użyliśmy zeskanowanego zdjęcia kwiatów (skaner EPSON GT6000), przykładowego rysunku z Adobe Fotoshop 2.5 oraz wektorowej ilustracji jabłka. Obrazek z kwiatami rzuca cień padający na zdjęcie pod spodem. Zastosowano tu trik, który polega na tym, że czarna separacja była puszczana osobno i tylko w niej istniał szary prostokąt pod zdjęciem. Separacji poddane zostało również zdjęcie dziewczynki z różnymi ustawieniami UCR/GCR.

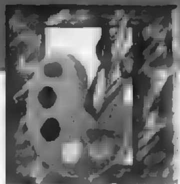
Ocenę separacji pozostawiam czytelnikom. Należy jednak wziąć pod uwagę, ze końcowa jakość zależy również od maszyny drukarskiej, farb, papieru i umiejętności drukarza, a te niestety w przypadku druku Amigowca nie są najwyższej jakości. ❑



COMPUGRAPHIC CG 9800 z RIP-em EMERALD		
kolor	kąt	gęstość
1200 dpi/133 Cpi		
Cyan	18,43	126,50
Magenta	71,57	126,50
Yellow	0,0	133,33
Black	45,0	141,42
2400 dpi/160 Cpi		
Cyan	18,43	157,789
Magenta	71,57	151,789
Yellow	0,0	160,0
Black	45,0	169,706

LINOTRONIC 300 z RIP-em 20		
kolor	kąt	gęstość
2400 dpi/133 Cpi		
Cyan	15,0037	138,142
Magenta	74,9963	138,142
Yellow	0,0	138,545
Black	45,0	138,158
2400 dpi/150 Cpi		
Cyan	15,0013	149,665
Magenta	74,9987	149,665
Yellow	0,0	149,412
Black	45,0	149,671

LINOTRONIC L 330R z RIP-em 30		
kolor	kąt	gęstość
1200 dpi/133 Cpi		
Cyan	18.43	157.53
Magenta	71,57	166,75
Yellow	0,0	158.75
Black	45,0	163,27
2400 dpi/160 Cpi		
Cyan	29,74	157,53
Magenta	66,80	166,75
Yellow	0,0	158,75
Black	45,0	163,27



la Vista, la Vista,

Krzysztof Trojanowski

Jak stworzyć animację w programie Vista Profesional v.3.5? Ten problem nurtuje wielu jego użytkowników. Odpowiedź na to pytanie nie jest jednak taka prosta.

Program Vista Pro umożliwia stworzenie płynnej animacji dowolnego landu (pliku z oznaczeniem .DEM) wybranego przez użytkownika programu. Wszystkie dane dotyczące scieżki, po której będzie przesuwała się kamera są z definiowane w specjalnych plikach tekstowych (SCRIPT).

Operacje z wykorzystaniem przykładów dołączonych do programu.

Jeżeli chcemy stworzyć animację dowolnego terenu postępujemy według następujących wskazówek:

a) Naciskając prawy klawisz myszki najeżdżamy na menu PROJECT. W menu tym wybieramy wielkość tworzonego projektu. Jeżeli posiadamy w naszej Amidze dość dużo pamięci wybieramy opcję SIZE-AUTO. SIZE-AUTO powoduje, że wielkość projektu będzie automatycznie zmieniana podczas wybierania preferencji dla określonego landu. Za pomocą tej opcji ustawiamy wielkość pola całego terenu. Im pole jest większe, tym większą drogę może pokonać nasza kamera w tym określonym terenie.

b) Następnie także z górnej listwy wybieramy menu LOAD, a z niego menu LOAD VISTA PRO DEM. Po ukazaniu się spisu gotowych projektów landów wybieramy dowolny plik z rozszerzeniem .DEM. Po chwili po środku lewej części ekranu ukaże się mapa terenu. (patrz Rysunek 1)

c) Z górnej listwy wybieramy menu SCRIPT. Z menu SCRIPT wybieramy CREATE. Ukaże się na ekranie spis scryptów. W oknie FILE wpisujemy nazwę naszego nowego scryptu i zatwierdzamy przez OK.

d) Jeżeli chcemy, aby nasza kamera poruszała się po lini prostej do punktu docelowego TARGET to:

- lewym przyciskiem myszy klikamy na opcję CAMERA (prawa górna strona ekranu) i ustawiamy na mapie terenu punkt, od którego ma zacząć się droga kamery. Naciskając opcje x, y, z, możemy zablokować poszczególne ruchy w układzie współrzędnych xyz. Jeżeli naciśniemy x, to ruchy kursorem po terenie nie będą miały wpływu na wartość współrzędnej x, jeżeli y, analogicznie nie będą zmieniały się wartości współrzędnej y, itd. Aby dokładnie us tawić nasz punkt w przestrzeni należy kliknąć lewym przyciskiem myszy na opcję P (obok TARGET i CAMERA - prawa górna część ekranu). Dwuwymiarowa mapa terenu zamieni się w przestrzenną. Po dokładnym ustawieniu punktu możemy ponownie wrócić do dwóch wymiarów przez powtórne kliknięcie na opcję P. (patrz Rysunek 2)

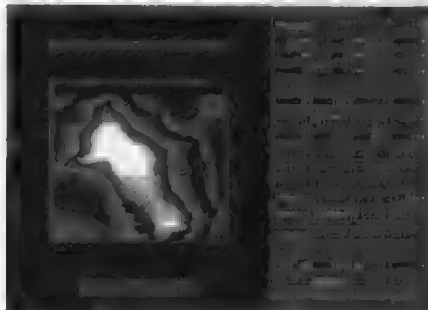
- po wybraniu punktu w którym ma się znajdować kamera, należy ustawić współrzędne punktu docelowego (do którego kamera będzie podążała). Lewym przyciskiem myszy klikamy na opcję TARGET obok opcji CAMERA i na mapie terenu ustawiamy miejsce, w którym zakończy się ruch kamery.

- po ustawieniu punktów na mapie terenu przechodzimy do generowania SCRYPTU ANIMACJI. W tym celu, prawym przyciskiem myszy klikamy na górną listwę i z menu SCRIPT wybieramy GENERATE. Po chwili na ekranie ukaże się spis scryptów. Program będzie chciał upewnić się czy nie zamierzamy zmienić nazwy tworzonego scryptu. Jeżeli nie to klikamy na OK. Następnie program zapyta się o liczbę kla tek animacji. Czym większa liczba klatek tym ruch będzie płynniejszy. Po wpisaniu liczby i zatwierdzeniu przez OK program zaczyna generować SCRYPT.

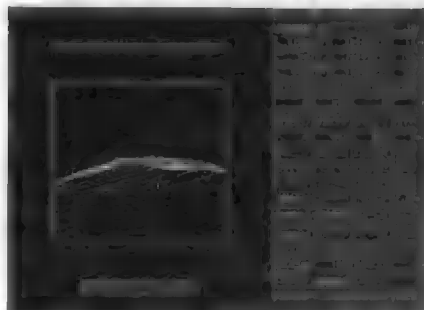
e) Jeżeli chcemy mieć pewność co do wyglądu ścieżki po której będzie poruszała się kamera, z menu SCRYPT wybieramy opcję PREVIEW. Następnie po ukazaniu się spisu scryptów wybieramy ten, który nas interesuje i klikamy na OK. Program zapyta się po chwili w jakiej formie chcemy oglądać kształt ścieżki - w dwuwymiarowej (2D) czy w przestrzennej (3D). Po wybraniu formy zobrazowwania program pokaże wygląd ścieżki we wszystkich klatkach animacji.

f) Jeśli chcemy, aby ruch kamery odbywał się po lini krzywej, należy postąpić następująco:

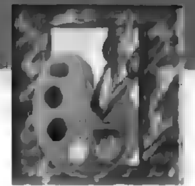
- kursory CAMERA i TARGET ustawić na mapie terenu we wspólnym punkcie,

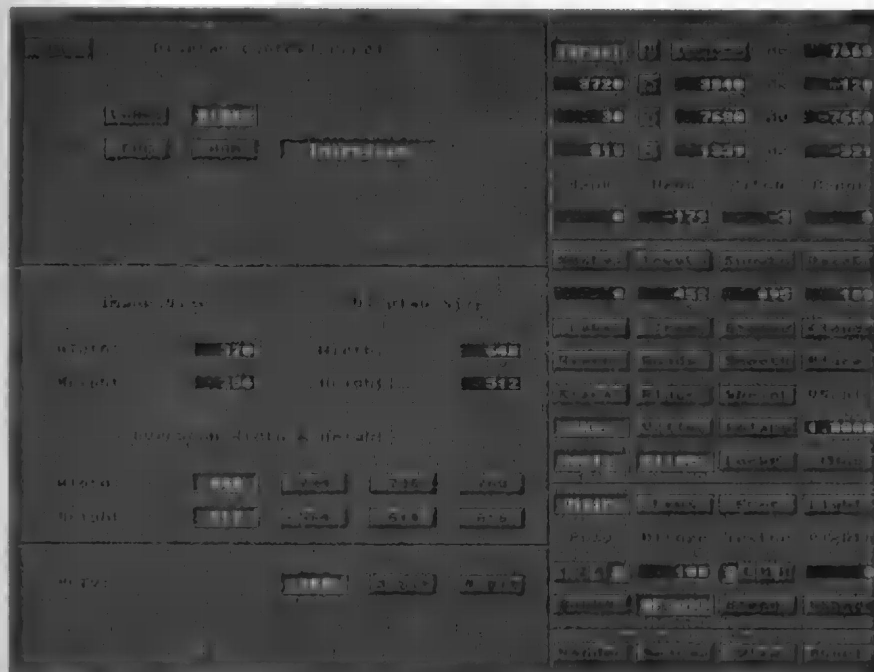


Rysunek 1



Rysunek 2





Rysunek 3

- następnie wybrać opcję ADD z menu SCRYPT,
- zmienić współrzędne dla TARGET (górna prawa część ekranu pod opcją TARGET),
- następnie ponownie wybrać opcję ADD i tak kolejno tworzyć poszczególne klatki animacji,
- w zmianie współrzędnych dla TARGET pomagają w znaczący sposób opcje HEAD i PITCH (prawa górna część ekranu). Opcja HEAD zmienia położenie w płaszczyźnie xy, a opcja PITCH w płaszczyźnie z.

Po utworzeniu SCRYPTU przechodzimy do generowania animacji. Prawym przyciskiem myszy wybieramy z górnej listwy menu GR MODES, a w nim opcję GRAPHICS PANEL. Po ukazaniu się ekranu z dostępnymi rozdzielczościami wybieramy tę, która nas interesuje. Przez wybranie opcji OK zatwierdzamy wszelkie zmiany i powracamy do menu głównego. (patrz Rysunek 3)

Następnie z menu SCRYPT wybieramy format zapisywanej na dysk animacji. Jeżeli chcemy, aby animacja była odtwarzana przez program Viewer 3, dołączony do VISTY v.3.0, wybieramy ANIMMODE - VANIM, jeżeli chcemy otrzymać pojedyncze klatki w 24 bitach, wybieramy ANIMMODE - IFF24.

Po ustawieniu rozdzielczości i formatu zapisywania animacji możemy przejść do generowania. Z menu SCRYPT wybieramy opcję EXECUTE. Po ukazaniu się spisu scryptów obecnych na dysku wybieramy, odpowiedni i zatwierdzamy przez OK. Następnie program zacznie generować animację według ustawionych przez nas parametrów.

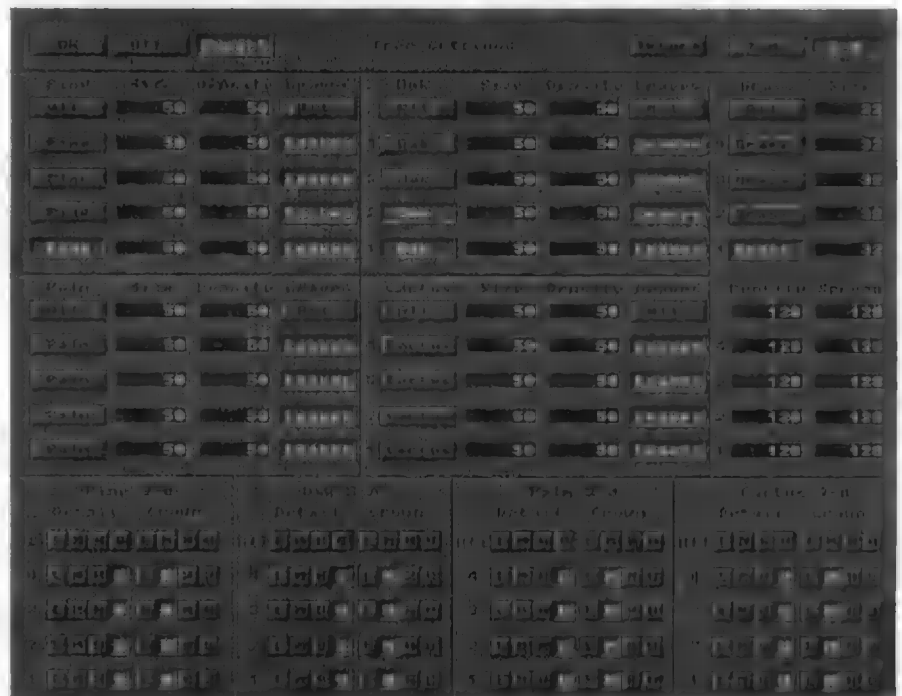
Gdy już nauczyliśmy się animować może zrodzić się inne pytanie. W jaki sposób modyfikować zawartość krajobrazu w programie Vista Profesional v.3.0?

Zajmijmy się definiowaniem miejsca występowania (na gotowym landzie) budynków, dróg, drzew oraz powierzchni trawiastych. Aby zacząć cokolwiek definiować musimy najpierw wgrać dowolny land z rozszerzeniem .DEM. W tym celu prawym przyciskiem myszy z górnej listwy wybieramy menu LOAD, a z niego opcję LOAD VISTAPRO DEM. Po ukazaniu się spisu landów istniejących na naszym dysku wybieramy ten który najbardziej się nam podoba i zatwierdzamy przez OK. Po ukazaniu się mapy terenu po lewej stronie ekranu, przechodzimy do definiowania miejsc występowania poszczególnych elementów krajobrazu. (patrz Rysunek 1.)

Opis menu odpowiedzialnego za preferencje krajobrazu.

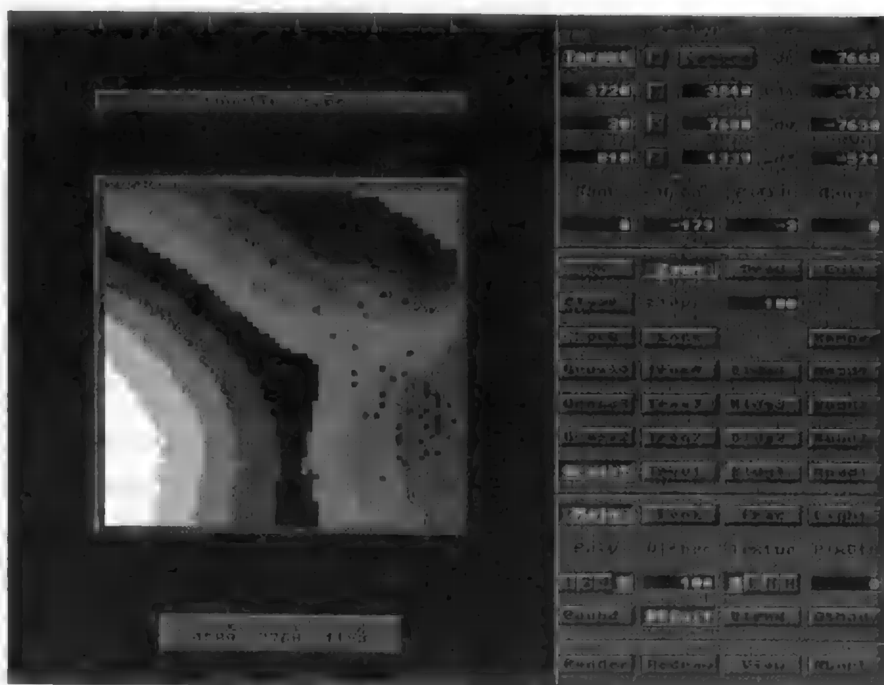
Po prawej stronie ekranu, pod menu odpowiedzialnym za ustawienie kierunku kamery, znajduje się menu preferencji krajobrazu. W menu tym znajdują się opcje, które są bezpośrednio odpowiedzialne za ostateczny wygląd tworzonego przez nas krajobrazu. Jeżeli chcemy, aby nasz wytwór zawierał odpowiednie elementy, musimy się zapoznać osobno z każdą opcją z tego menu.

Od góry:
- SeaLvl - służy do definiowania poziomu morza,
- TreeLn - wysokość na jakiej będą znajdowały się drzewa,
- SnowLn - wysokość na jakiej będzie występował śnieg,



rysunek 5





Rysunek 4

- Haze Dn - wysokość na jakiej będzie znajdowała się mgła,

- Lake - pozwala zdefiniować poziom na którym będzie znajdowało się jezioro,

- Tree - pozwala na umieszczenie drzew w krajobrazie. Poprzez dwukrotne kliknięcie na tej opcji możemy zdefiniować rodzaj drzew, (patrz Rysunek 4)

- Stereo - opcja ta umożliwia, poprzez odwrócenie obrazu pola i widoku kamery, stworzenie efektu trójwymiarowości (potrzebne specjalne okulary),

- Clouds - pozwala na umieszczenie chmur w naszym krajobrazie. Poprzez dwukrotne kliknięcie na tej opcji możemy ustawić rodzaj zachmurzenia.

- River - pozwala na umieszczenie rzeki w naszym krajobrazie,

- Roads - pozwala na umieszczenie dróg,

- Smooth - wygładza granice terenu,

- Place - umożliwia zaznaczenie miejsca w którym będą występowały drogi, budynki, drzewa i powierzchnie trawiaste, (patrz Rysunek 5)

- Stars - umieszcza gwiazdy na niebie,

- Bldgs - umieszcza budynki,

- Shrink - anuluje ostatnio wykonaną operację (dotyczy tylko menu odpowiedzialnego za preferencje krajobrazu).

- Sky - umieszcza niebo,

- Valley - umieszcza kotliny,

- Enlarg - pozwala na powiększenie określonego obszaru terenu,

- Horiz - umieszcza linię horyzontu w naszym krajobrazie,

- Cliffs - tworzy skaliste powierzchnie, urwiska,

- LockP - blokuje paletę przy tworzeniu animacji,

- CMap - za pomocą tej opcji można zdefiniować paletę kolorów wykorzystywaną przy tworzeniu krajobrazu. Wybierając prawym przyciskiem myszy z górnej listwy ekranu opcję LOAD CMAP, możemy wczytać z dysku gotowe palety. Można w ten sposób zmienić porę roku, jaką będzie przedstawiał tworzony przez nas krajobraz (wiosna, lato, jesień, zima),

- VScale - przedstawia mapę terenu w różnej zdefiniowanej przez nas skali.

Umieszczanie drzew, budynków, dróg i powierzchni trawiastych.

Po zapozaniu się z opcjami menu dotyczącego ustawienia preferencji krajobrazu przechodzimy do umieszczania drzew, budynków, dróg, powierzchni trawiastych. Klikamy lewym klawiszem myszki na opcję PLACE znajdującą się w prawej środkowej części ekranu. Po ukazaniu się nowego menu klikamy lewym przyciskiem myszy na opcję ZOOM i zaznaczamy, najeżdżając myszką na mapę terenu, obszar w którym chcemy umieścić drzewa, budynki, drogi i powierzchnie trawiaste. Jeżeli chcemy umiejscowić powierzchnie trawiaste wybieramy GRASS 1, to najeżdżając na mapę terenu klikamy lewym przyciskiem myszy w miejscu, w którym ma ona występować. (patrz Rysunek 5.) Numeracja od 1 do 4 przy GRASS, ROAD, BLDG, TREE oznacza rodzaj pojedynczego elementu (trawy, drogi, drzewa) np. pod opcją TREE 1 mogą znajdować się palmy, a pod opcją TREE 2 kaktusy. Opcją CLEAR możemy wyczyścić wszystkie naniesione na mapę terenu elementy. Opcja DRAW auto matycznie zaznacza nam elementy wykorzystywane w danym krajobrazie. REMOVE kasuje zaznaczone pola.

Jeżeli skończyliśmy już rozmieszczanie dróg, budynków, drzew klikamy lewym przyciskiem myszy na OK i powracamy do menu głównego. Klikamy dwókrotnie na opcję TREE. Na ekranie ukaże się nowe menu. W menu tym definiujemy rodzaj drzew, gęstość i wielkość powierzchni trawiastych. (patrz Rysunek 4.) Do wyboru mamy cztery gatunki drzew: sosna (PINE), dąb (OAK), palma (PALM), kaktus (CACTUS).

Dla każdego gatunku możemy ustawić: wielkość, liczbę drzew (SIZE), gęstość (DENSITY), dokładność rysowania korony drzew (CROWN), dokładność odwzorowania szczegółów (DETAIL).

Przy tych ostatnich dwóch mamy do wyboru opcje L, M, H, U. Oznaczają one stopień dokładości L (najniższy), U (najwyższy). Ustawienie najwyższego stopnia U (ULTRA) jest konieczne tylko wtedy, gdy nasze drzewa będą znajdowały się blisko kamery.

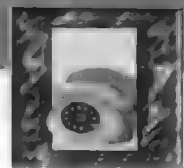
Możemy także za pomocą opcji LEAVES włączyć lub wyłączyć ulistnienie drzew. Dla powierzchni trawiastych dodatkowo za pomocą opcji SPREAD ustawiamy wielkość obszaru na którym rośnie trawa.

Ustawianie poziomu morza i wysokości na której występują drzewa, śnieg i mgła.

Jeżeli chcemy ustawić poziom, na którym będzie występowało morze, lewym przyciskiem myszy klikamy na opcję SEALVL po prawej stronie ekranu. Następnie najeżdżamy na mapę terenu i klikamy na dowolny obszar. Warto wiedzieć, że obszary zaznaczone kolorem białym są położone najwyżej, a ciemnym zielonym najniżej.

Te same czynności powtarzamy przy ustawianiu wysokości na której będą występowały drzewa, śnieg i mgła.

Mam nadzieję, że moje krótkie lekcje pozwolą Wam na lepsze wykorzystanie możliwości programu Vista Pro 3.5. Czekam na listy z propozycjami tematów do opisania. Do zobaczenia. ❑



Dlaczego TERM?

Tomasz Matajek

Wśród wielu użytkowników komputerów osobistych utarło się przekonanie, że dobry program musi kosztować co najmniej kilkaset dolarów. H■ szczęście nie jest to prawda. Często zdarza się, iż drogie programy komercyjne nie ■■ warte nawet ułamka swojej ceny, natomiast programy typu Public Domain, Free Ware lub Shareware mogą być znakomite.

Przykładem jest właśnie Term - doskonały program telekomunikacyjny. Nie ustępuje on możliwościami komercyjnym programom tego typu, ■ pod wieloma względami znacznie je przewyższa. Poniżej podanych zostanie kilka praktycznych wskazówek, jak najszybciej zaprząc Terma do pracy, bez konieczności czytania całej instrukcji obsługi liczącej ponad 200 KB. (Nawiasem mówiąc instrukcja ta jest przejrzyście napisana, ■ w dodatku jest w formie "Amiga Guide"- przyjemnej w użyciu i coraz bardziej ostatnio popularnej).

Instalacja programu jest prosta. Na dysku roboczym należy stworzyć directory o nazwie "Term". Tam też należy skopiować odpowiednią do posiadanego procesora wersję programu (istnieją wersje na 68000, 68010, 68020 i 68030). Wskazane jest utworzenie wewnątrz tej szuflady dwóch nowych katalogów, służących następnie do przyjmowania i odbierania danych (patrz punkt 3). Ostatnią czynnością jest dopisanie do "user-startup" następującej linii:
"assign TERM: <sciezka określająca gdzie jest szuflada z programem>"
na przykład: assign TERM: dh0:term.
W tym momencie polecam wykonać reset, gdyż powyższe przypisanie dopiero wtedy stanie się aktywne.

Początkujący użytkownik modemu będzie zapewne nieco przestraszony dużą ilością opcji dostępnych w tym programie. Na szczęście, tylko niektóre parametry muszą zostać zmienione, pozostałe ustawione są od razu tak, aby można było zacząć pracę nie troszcząc się zbytnio ■ ich ustawienie. Do rozpoczęcia zabawy wystarczy na razie skupić się na kilku punktach z menu "Settings". Najistotniejsze punkty z tego menu zostaną kolejno omówione.

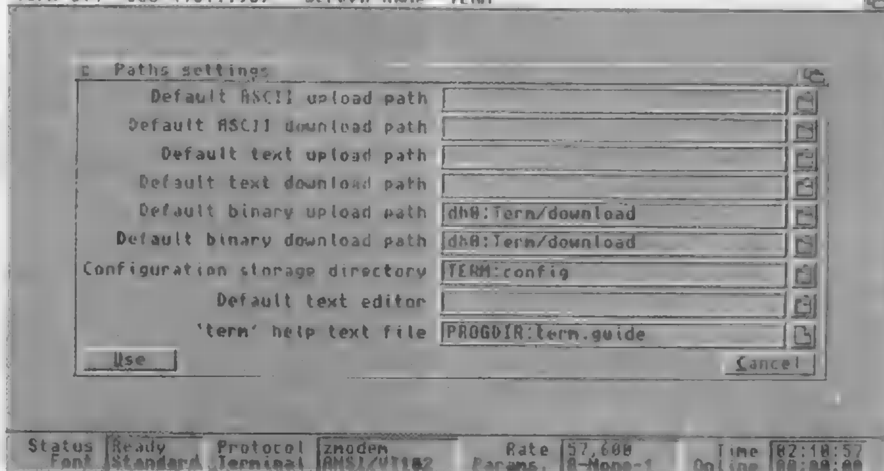
1. Serial

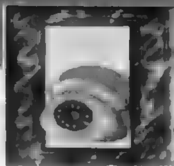
- Baud rate.
Wielkość ta ustala prędkość z jaką modem będzie komunikować się ■ komputerem. Nie może ona być mniejsza, niż maksymalna prędkość z jaką dany modem jest w stanie przesyłać dane do innego modemu. Dla szybkich modemów (9600 i 14400) ■ kompresją danych 4:1, szybkość ta powinna wynosić odpowiednio 38400 i 57600, aby w pełni wykorzystać zalety kompresji. W praktyce przesyła się jednak pliki już skompresowane - nie są one ponownie kompresowane przez modem, dlatego prędkość 19200 przeważnie wystarcza.
Użytkownicy poczciwej Amigi 500 powinni ustalić tę wartość na 19200. Ustawienie wyższej wartości nie ma sensu, gdyż pojawiać się będą błędy transmisji na linii komputer - modem.
Posiadacze Amig 1200 mogą wybrać 38400, natomiast dla A3000 i A4000 można bezproblemowo ustawić 57600.

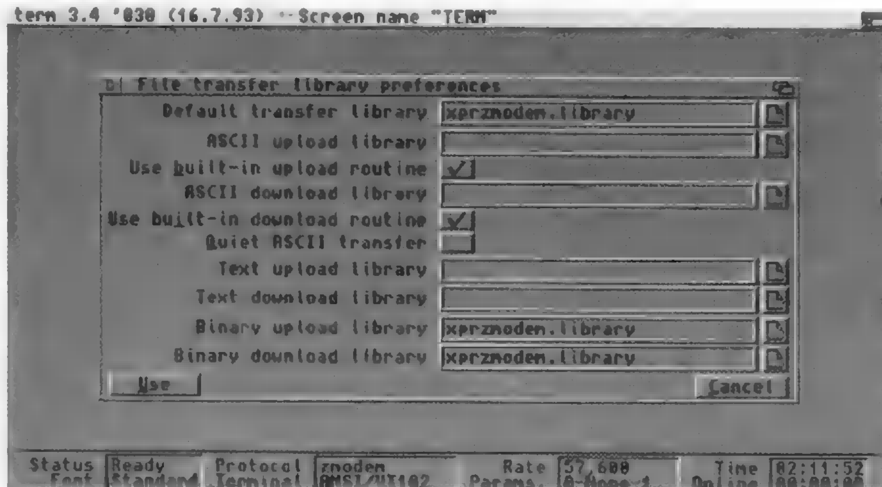
- Bits/char.
Parametr ten mówi ile bitów przypada na jeden znak. Należy wybrać "Eight".

- Parity.
Określa, czy w przesyłanych danych ma być użyty bit kontroli parzystości. Wybieramy "None" (brak bitu kontroli parzystości).

- Stop bits.
Ustala liczbę bitów stopu. Wybieramy 1.







Trzy powyższe parametry transmisji są najczęściej spotykane w praktyce, a ustawienie to zwane jest w skrócie 8-N-1.

- Handshaking.
Jest to używany przez komputer rodzaj potwierdzenia przesłania pakietu danych. Najszybszy i najskuteczniejszy (bo hardwareowy) jest rodzaj RTS/CTS lub jeszcze lepiej, bardziej od niego rozbudowany RTS/CTS (DSR).

- Duplex.
Przeważnie ustawia się "Full", ale w niektórych przypadkach lepsze jest ustawienie "Half (local echo)".

- Buffer size.
Ustala wielkość bufora do którego przesyłane są odbierane lub transmitowane dane. Wielkość 16384 jest absolutnie wystarczająca jeśli transmisja odbywa się z lub na twardy dysk. Gdy chcemy odbierane dane zapisywać na dyskietki, dobrze jest ustawić duży bufor - najlepiej taki, jak wielkość przesyłanego pliku. Ma to na celu wyeliminowanie przerw w transmisji związanych z wolnym zapisem na dyskietki.

Z gadgetów w drugiej kolumnie ustawiamy tylko "Check carier", natomiast w gadżecie "Serial device" wybieramy aktualnie używany driver (wystarczy kliknąć na gadżet i wybrać odpowiedni driver myszką). Standardowo dla portu szeregowego Amigi jest to dobrze znany "serial.device". Dla tych, którzy mają karty z dodatkowymi portami szeregowymi, istnieje możliwość wyboru portu przez wpisanie jego numeru do gadżetu "Unit number". Normalnie jest on ustawiony na zero. Parametr "Quantum" nie jest dla nas w tej chwili interesujący - zostawiamy go tak jak był, czyli 256.

Dla potwierdzenia ustawionych parametrów wybieramy "Use".

2. Transfer.

Aby ustandaryzować proces przesyłania danych, powszechnie stosuje się tzw. protokoły transmisji. Jest ich wiele, mają różne wady i zalety, ale najpopularniejszym i chyba obecnie najlepszym z nich jest protokół ZModem. Stosuje się go do przesyłania wszelkich danych, niezależnie od ich rodzaju (np. programy, teksty czy grafiki traktowane są jednakowo).

- Default transfer library.
Wybieramy (wpisując bezpośrednio lub klikając na gadget) xprzmodem.library.

- Binary upload library.
Wybieramy jak wyżej.

- Binary download library.
Wybieramy jak wyżej.

Ponadto należy zaznaczyć opcje:
- Use built-in upload routine,
- Use built-in download routine.

Pozostałe gadżety, ze względu na wybrany protokół, nie są dla nas istotne.

Ponownie, dla potwierdzenia ustawionych parametrów wybieramy "Use".

3. Paths.

W tym punkcie będą dla nas istotne tylko trzy gadżety:

- Default binary download path.
Służy do ustawienia scieżki odbieranych danych (np. dh0:term/download).

- Default binary upload path.
Służy do ustawienia scieżki wysyłanych danych (np. dh0:term/upload).

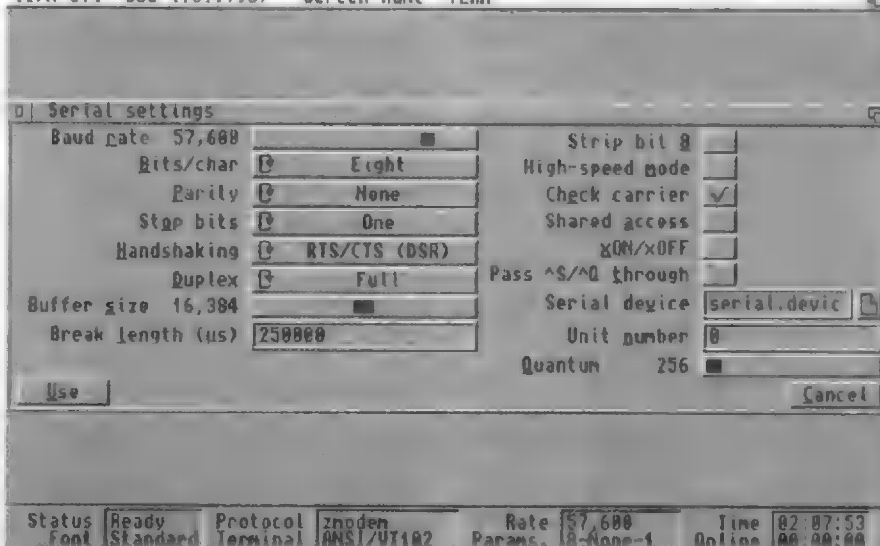
- Configuration storage directory.
Umożliwia ustalenie miejsca na dysku, w którym będzie zapisana konfiguracja. Dla własnej wygody powinno się tu wpisać "Term:config".

Potwierdzamy nasz wybór wybierając "Use".

4. Save settings.

Trzeba wybrać ten punkt, aby nowe ustawienie zostało zapamiętane [redacted] przyszłość.

Tak skonfigurowany terminal jest gotowy do pracy. Używając opcji "Dial phone number" z menu "Phone" dzwonimy do najbliższego BBSu i ... zaczynają się kolejne problemy. Ale to już nie jest temat tego artykułu... ❑





Amigowe muzykowanie...

Marcin Gackowski

Witam wszystkich w nowym dziale Amigowca, poświęconym muzyce. Chciałbym, aby rubryka ta stała się czymś w rodzaju poradnika dla muzyków tworzących na Amidze. Jeśli więc, Drogi Czytelniku masz jakiś problem natury technicznej, opisz go w miarę sensownie i prześlij na adres redakcji, a ja, w miarę możliwości, postaram się go rozwiązać. Nie jestem, co prawda, wszechwiedzącym w tej dziedzinie, ale uważam, że wiem co nieco i postaram się to wykorzystać.

Nie skrywam także, że pierwsze odcinki moich rozważań poświęcę głównie Protrackerowi, gdyż na nim tworzę i go znam najlepiej. Oczywiście pytania typu: "Jak zrobić dobry moduł?" pozostaną bez jednoznacznej odpowiedzi. Mam jednak nadzieję, że lektura moich artykułów ułatwi zrozumienie na czym polega tworzenie muzyki na komputerze i sprawi, że tworzone przez Was moduły będą coraz to lepsze.

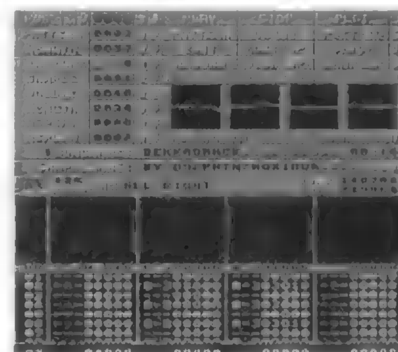
Na początek chciałbym napisać parę słów o samplach, czyli instrumentach. Sample, jakie są, każdy słyszy. Czyste lub szumiące, oryginalne bądź oklepane. Najczęściej zaś - kradzione. Tak, nie ma się co oszukiwać. Kradniemy sample skąd popadnie. Często z dem, w których często jest dokładnie napisane, że to muzyk jest właścicielem sampli i po tym następuje wykaz instrumentów (zazwyczaj klawiszowych), które ów muzyk wkorzystał. Dlaczego więc używając cudzych sampli tak rzadko zaznaczamy skąd one pochodzą? Może uważamy, że zmiana głośności sampla, czy też jego wzmocnienie czyni nas jego kreatorem. Bzdura. A przecież korzystanie z cudzych sampli nie jest niczym złym. Tym bardziej u nas, gdzie nie każdego stać na syntezator. Korzystajmy więc z cudzej pracy, ale czyńmy to z należnym szacunkiem. Niełatwo mi to pisać, bowiem sam, zazwyczaj, nie postępowałem w wyżej opisany sposób. Jednak dla przykładu postanowiłem podawać w modułach autora sampli, jeżeli jest mi on znany, oczywiście.

Skąd brać sample? To kluczowy problem początkujących muzyków. Można wybrać się na giełdę, gdzie najprawdopodobniej handlarz po kilku "fachowych" zdaniach zaproponuje nam kilka dysków serii ST-XX (gdzie XX, zazwyczaj, nie przekracza dziesięciu). Na początek wystarczy. Jednak po pewnym czasie dojdziemy do wnisku, że moduły stworzone z tych sampli brzmią nieciekawie i tak jakoś dziwnie. Dyski serii ST, a w szczególności te najbardziej rozpowszechnione (czyli numery od 1 do 10), zawierają brzmienia ubogie i bardzo sztuczne, kiedyś zachwycające, dziś po prostu śmieszne.

Co więc robić? Można wybrać się na najbliższe Party, gdzie na pewno znajdzie się jakiś uprzejmy muzyk, od którego nieodpłatnie będziemy mogli przegrać kilkanaście dysków z interesującymi brzmieniami (swego czasu Mr. Root rozdawał sample na Party w Gdyni). No i problem z głowy.

A jeżeli w najbliższej przyszłości nikt nie organizuje Party? Można spróbować własnych sił w samplowaniu. Należy zaopatrzyć się w sampler tzn. kupić bądź pożyczyć np. od kodera jakieś grupy. Koder ma zazwyczaj sampler który służy mu do pożyczania. Potrzebne też będzie źródło dźwięku (instrument, magnetofon, odtwarzacz CD, ewentualnie mikrofon + głos) oraz dużo cierpliwości. Trzeba naprawdę wiele eksperymentować, aby wynik był warty nagrania na dysku. Z doświadczenia wiem, że najłatwiej samplować instrumenty perkusyjne i wszelakie efekty (z głosem ludzkim na czele), zaś najtrudniej te, które miałyby posłużyć nam za instrument prowadzący. Od razu odradzam notorycznego używania zsamplowanych całych sekwencji np. perkusji. Lepiej poświęcić trochę czasu i "pociąć" długiego sampla na krótsze. Dlaczego? Nie tylko dla oszczędności pamięci. Jeden długi sampl jest bardzo charakterystyczny i użycie go w paru kompozycjach będzie autoplagiatem. Natomiast z krótkich sampli, stworzonych z długiej sekwencji, możemy ułożyć kilkanaście różnych podkładów perkusyjnych. Zresztą dokładniej samodzielnym samplowaniem zajmę się w następnym numerze, gdyż uważam, że warto opanować tę nietrudną sztukę. A i efekty czasami są zadziwiajce.

A co począć, gdy nie mamy samplera? Pozostaje nam rippowanie, czyli



Ekran ProTrackera



wyciąganie sampli z dem, gier i programów. Jest to najbardziej popularny sposób na zdobycie instrumentów. Najłatwiej wyciągnąć sample za pomocą Action Replay'a. Najtrudniej (szczególnie z nowszych produktów), korzystając z programowych ripperów. Spowodowane jest to tym, że koderzy stosują, coraz częściej, swoje własne procedury do odtwarzania muzyki i ripper "głupieje", nie wiedząc z jakiego formatu muzyką ma do czynienia. Pozostaje wtedy zgranie pamięci na dysk i "przesłuchiwanie" jej na np. Audiomasterze. Potem to już tylko żmudne wycinanie i zgrywanie [redacted] dysk każdego instrumentu. Jeżeli będzie zapotrzebowanie ze strony Czytelników na dokładny (krok po kroku) opis rippowania, życzenie zostanie spełnione.

A teraz kilka rad dla początkujących posiadaczy kilku dyskietek z samplami:

Przed przystąpieniem do pracy dobrze jest posegregować sample. Można rozdzielić sample do odpowiednich katalogów (np. gitary do katalogu guitars, instrumenty perkusyjne do katalogu drums) lub rozdzielić sample brzmieniowo na oddzielne dyskietki i nadać nazwy stosowne do zawartości (np. chórki na dysk Choirs, różne efekty na dysk Effects). Dozwolone są dowolne sposoby segregacji. Szczególnie polecam sposób, gdzie wszystkie sample wyciągane z jednego dema są nagrywane [redacted] tym samym (powiedzmy trzyliterowym) identyfikatorem. Dzięki temu uzyskujemy na dysku posegregowane grupy sampli pasujących do siebie brzmieniowo. Oto przykładowy katalog ułożony dzięki stosowaniu tej metody segregacji:

FER-BASS-1
FER-BASS-2
FER-DRUMS-KICK
FER-DRUMS-HIHAT
FER-DRUMS-SNARE
FER-EFFECT-WIND
MEL-CHOIR-DUR
MEL-CHOIR-MOLL
MEL-STRINGS-1
MEL-STRINGS-2

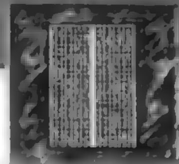
Myślę, że widać o co tu chodzi. Jest to, moim zdaniem, najlepszy sposób i stosuję go od dawna.

Niezwykle ważnym elementem jest właściwe (tematyczne) nazewnictwo sampli. Gdy widzę dyski z samplami nazwanymi 1,2,3,4,5... lub Bum,Bęc, Trach..., zaczyna mi się robić niedobrze. Sample powinny być w miarę odpowiednio nazwane. Można nazywać je po angielsku, polsku lub chińsku ale tak, aby ktoś oprócz nas wiedział co jest [redacted] dysku. Dobrym sposobem na nazwy jest zapożyczanie ich z syntezatorów, chociaż trzeba przyznać, że dla niektórych sampli trudno znaleźć odpowiednią nazwę. Parę przykładów na określenie brzmień perkusji: Bassdrum, Kick, Stopa, Snare, Werbel, Hihat itp. To naprawdę ułatwia pracę, szczególnie przy kilkudziesięciu dyskach z samplami.

Jeżeli decydujemy się na korzystanie z PLST, pamiętajmy o czytelnym oznaczaniu dysków takimi samymi symbolami jak w edytorze PLST.

Tyle słowem wstępu o samplach. Konkretne pytania proszę kierować na adres redakcji.

Życzę owocnej pracy. Za miesiąc dalszy ciąg "programu". ❑



Cicha woda brzegi rwie...

Jarosław Chrostowski

Każdy z nas to zna: ciągłe wstukiwanie tekstów, znęcanie się nad myszą i jakąś grafiką, łamanie joysticka (wiem coś na ten temat, parę już mi się w ręku rozpadło) czy wyciskanie siódmych potów z procesora podczas dwudziestej próby ustawienia dobrego oświetlenia w Imagine...

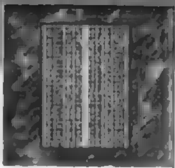
Gdy jednak nie uruchomiliśmy żadnego "procesorożernego" programu, a ręce odpoczywają od klawiszy i myszy (lub one od nas), wtedy wydaje się, że Amiga śpi snem niewiniątka - po prostu leni się. Nic bardziej mylącego. Jak się za chwilę przekonamy, jest to spokój pozorny. W pamięci naszego komputera - nawet wtedy, gdy "nic nie robi" - dzieje się wiele ciekawych rzeczy.

Jedną z największych zalet systemu operacyjnego Amigi jest multitasking. Rzecz sama w sobie wcale nie nadzwyczajna, możliwość równoczesnej pracy wielu programów oferuje np. UNIX. Jednak fakt, że system operacyjny tak małego komputera (mam na myśli domowe Amigi - 500, 600, 1200 itp.) oferuje taką możliwość jest sam w sobie rewelacją. Niestety, wiele osób nie jest w stanie docenić zalet pracy w multitaskingu, bowiem mała ilość pamięci (1 MB RAM to doprawdy malutko) nie pozwala na uruchomienie większej ilości programów. Poza tym praca "na krawędzi" pamięci z reguły nie należy do przyjemnych, bowiem system będzie co chwilę o tym fakcie przypominał. Jeśli do tego dodamy, że znaczna część programów jest napisana niepoprawnie lub złamana przez jakiegoś domorosłego hackera - nie czarujmy się, większość dostępnych na rynku polskim programów na Amigę to kopie pirackie, sprawy nie załatwia kilkanaście rozprowadzanych gier - wtedy okazuje się, że na dodatek system co chwilę zawiesza się. Wierzcie mi jednak, że multitasking w zastosowaniach użytkowych to narzędzie niedoceniane tylko przez tych, którzy z niego nie mogą korzystać.

Połączone wysiłki firm Commodore, Motorola i Microsoft doprowadziły do powstania unikatowego systemu operacyjnego. System ten pozwala na uruchamianie programów jako odrębnych zadań - tzw. tasków (multitasking w tłumaczeniu na polski to właśnie wielozadaniowość). Tak naprawdę task w systemie Amigi ma dość ograniczone możliwości, m.in. nie można w nim korzystać z poleceń związanych z DOSem. Sprawę rozwiązuje możliwość uruchomienia programu jako tzw. procesu, którego dany task jest częścią składową. Tu jednak chciałbym się przede wszystkim skoncentrować na pojęciu tasku - ramy tego artykułu są i tak o wiele za małe na szczegółowy opis.

Najpierw może parę słów o ogólnych mechanizmach zarządzania taskami. Otóż w danej, konkretnej chwili procesor może wykonywać tylko jeden rozkaz programu, tak więc tak naprawdę ZAWSZE w pamięci działa jednocześnie... tylko jeden program. Cała sztuka polega na tym, że różne programy w różny sposób obciążają procesor. Np. edytory tekstu zajmują czas procesora praktycznie tylko w momencie otrzymywania sygnałów z klawiatury lub myszy, wszystkie inne operacje (typu wyrównywanie tekstu) również nie trwają zbyt długo. Inaczej sprawa się ma z programami wykonującymi obliczenia, np. uruchomienie procesu obliczania jakiegoś rysunku w programie typu ray-tracing obciąża procesor w znacznym stopniu. Jednak bez specjalnego zwolnienia tych obliczeń możemy uruchomić obliczenia w Imagine i jednocześnie korzystać z Ceda do napisania listu. Ten ostatni bowiem będzie zajmował procesor mniej więcej z taką częstotliwością, z jaką stukamy w klawisze, na dodatek dość krótko. Efekt jest wyśmienity - oba programy pracują "jednocześnie" z taką samą szybkością (kto będzie się kłócił o kilkanaście sekund opóźnienia przy obliczeniach trwających godzinami). Trochę gorzej sprawa by się miała, gdybyśmy uruchomili dwa programy Imagine i w każdym rozpoczęli obliczenia. Jak łatwo się domyślić, procesor raz wykonywałby fragment jednego programu, po czym od razu otrzymywałby fragment drugiego. Efekt - dwa razy wolniejsze obliczenia, co w efekcie odpowiada obliczeniu jednego, a potem drugiego obrazka na tym samym Imagine.

Jak widać, multitasking to ciągłe przełączanie procesora pomiędzy programami - raz wykonujemy fragment jednego programu, raz drugiego. Oby odciążyć procesor większość programów żąda dostępu tylko wtedy, gdy się wydarzy coś ciekawego, np. wciśniemy klawisz myszy i Ced musi wyświetlić nowy znak na ekranie. Przykłady te pozwalają zauważyć, że istnieją trzy typy tasków (jest parę więcej, lecz te są najważniejsze). Pierwszy typ, tzw. RUNNING, task, który jest aktualnie wykonywany przez procesor. Z oczywistych względów zawsze istnieje tylko jeden task w tym stanie. Następny typ to taski, które chcą się dostać do procesora, lecz ten jest zajęty wykonywaniem innego (patrz przykład Imagine - Imagine). Ten typ nosi nazwę READY i oznacza task, który chce się dostać do zajętego procesora. Ostatnim typem jest WAITING, oznaczający taski, które oczekują na



informację z zewnątrz (np. Ced czekający na sygnał z klawiatury) i dopóki ona nie nadejdzie, ani im w głowie (w tasku?) dopychać się do procesora.

Wszędzie tam, gdzie jest tłok, a chce się uniknąć chaosu, pojawia się kolejka. Efektem pojawienia się kolejki jest, jak wiemy, efekt pojawienia się listy. Amiga zachowuje się tu bardzo podobnie, tworząc dwie listy tasków: w stanie READY oraz w stanie WAITING. Brak listy RUNNING jest chyba oczywisty. Każdy z tasków zawiera informację o tym, gdzie znajduje się poprzedni oraz następny task. Jeśli nie ma wcześniejszego tasku (czyli mówimy o pierwszym tasku w liście), wtedy zamiast wskaźnika na task mamy wartość 0. Podobnie ostatni element listy ma wskaźnik na następny task ustawiony na 0. Taki mechanizm tworzenia list jest podstawą działania systemu operacyjnego Amigi i pozwala na bardzo proste jej modyfikowanie, czyli dokładanie i usuwanie tasków (wystarczy zmienić tylko wskaźniki we wcześniejszym i w późniejszym tasku).

Wszystkie podstawowe informacje o systemie Amigi (konkretnie o Execu) zawiera struktura ExecBase. Jedna ze składowych tej struktury to wskaźnik:

```
struct Task *ThisTask;
```

który zawiera adres aktualnie wykonywanego tasku. Poszczególne taski nie wiedzą, kiedy mogą rozpocząć pracę na procesorze, sprawdzają więc wartość tej składowej. Jeśli zawiera ona wskaźnik na dany task, wtedy task ten zaczyna być przetwarzany przez procesor. W takiej sytuacji task ten jest oczywiście usuwany z listy WAITING lub READY i otrzymuje status RUNNING. Jednak wykonywanie tasku może zostać w każdej chwili przerwane, bowiem w strukturze ExecBase składowa ThisTask może zostać zmieniona i wskazać na inny task. Wyrzucony task jest wstawiany w listę READY lub (w przypadku, gdy w przydzielonym czasie zdążył wykonać swoją robotę i oczekuje na jakiś komunikat) w listę WAITING. Zarządzaniem taskami zajmuje się procedura systemowa Switch(), przełączająca je 50 razy na sekundę (tj. z częstotliwością odpowiadającą przerwaniu generowanemu w chwili osiągnięcia przez strumień elektronów monitora pierwszej, górnej linii ekranu). Aby uniemożliwić przełączanie tasków należałoby więc zatrzymać przerwania funkcją Disable() (ponowne włączenie następuje dzięki wywołaniu procedury Enable()).

Powstaje pytanie, jak długo jeden task może zajmować procesor. Czas ten jest mustalany na podstawie wartości priorytetu tasku - im wyższy priorytet, tym więcej czasu procesora zajmie dany task. Czas dostępu do procesora zależy również od ilości tasków w stanie READY, tak więc task z określonym priorytetem będzie zajmował procesor dłużej, gdy lista READY jest krótka, krócej, gdy składa się z wielu tasków, zawsze jednak task o wyższym priorytecie zajmie procesor dłużej, niż task o priorytecie niższym. Ten mechanizm zarządzania taskami powoduje, że nawet task z bardzo niskim priorytetem uzyska dostęp do procesora. Wartość priorytetu może mieścić się w zakresie od -128 do 127, przy czym większość tasków uruchamianych w systemie ma wartości od -20 do 20.

Każdy z tasków ma przydzielony swój stos, a więc obszar pamięci, w którym znajdują się dane robocze tasku (m.in. w momencie przeskoczenia do następnego tasku na stos poprzedniego tasku odkładane są wartości poszczególnych rejestrów procesora). Chcąc uzyskać ogólną informację o tasku, oprócz jego nazwy, priorytetu oraz sygnałów, na które oczekuje (o tym za chwilę) dobrze byłoby podać, gdzie w pamięci rozpoczyna się stos tasku oraz jaki jest jego rozmiar. Zobaczmy jak to zrobić.

Jak wspomniałem, uruchomienie programu jest związane z pojawieniem się w systemie tasku. Task taki opisuje struktura typu Task następującej postaci:

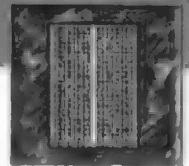
```
struct Task
{
        struct Node     tc_Node;
        UBYTE           tc_Flags;
        UBYTE           tc_State;
        BYTE            tc_IDNestCnt;
        BYTE            tc_TDNestCnt;
        ULONG           tc_SigAlloc;
        ULONG           tc_SigWait;
        ULONG           tc_SigRecvd;
        ULONG           tc_SigExcept;
        UWORD           tc_TrapAlloc;
        UWORD           tc_TrapAble;
        APTR            tc_ExceptData;
        APTR            tc_ExceptCode;
        APTR            tc_TrapData;
        APTR            tc_TrapCode;
        APTR            tc_SPReg;
        APTR            tc_SPLower;
        APTR            tc_SPUpper;
        VOID            (* tc_Switch) ();
        VOID            (* tc_Launch) ();
        struct List tc_MemEntry;
        APTR            tc_UserData;
};
```

Nie chciałbym zajmować się szczegółowym opisem tej dość skomplikowanej struktury, stąd przyjrzymy się bliżej jedynie kilku składowym. Pierwszą składową jest tc_Node, będąca strukturą typu Node. Struktura Node to jedna z najważniejszych struktur systemu Amigi. Służy ona do organizowania list i wygląda następująco:

```
struct Node
{
        struct Node *   ln_Succ;
        struct Node *   ln_Pred;
        UBYTE   ln_Type;
        BYTE    ln_Pri;
        BYTE *  ln_Name;
};
```

Składowa ln_Succ wskazuje na następną pozycję w liście (lub jest wyzerowana w przypadku ostatniego elementu listy), składowa ln_Pred to wskaźnik na poprzedni element listy (lub zero w przypadku pierwszego elementu). Ponieważ struktura Node jest wykorzystywana w wielu miejscach systemu, istnieje więc konieczność rozróżnienia, czy jest ona aktualnie elementem listy tasków, czy jakiejś innej listy. Typ listy określony jest przez składową ln_Type, dla listy tasków jest to wartość NT_TASK. Składowa ln_Pri zawiera priorytet (w naszym przypadku jest to priorytet tasku), natomiast ln_Name wskazuje na nazwę (tutaj jest to wskaźnik na nazwę tasku).

Składowe o nazwach tc_SPLower i tc_SPUpper zawierają odpowiednio adres początku i końca stosu przydzielonego danemu taskowi. Aktualną pozycję w stosie wskazuje tc_SPReg. Ponieważ stos jest zapełniany "od góry", więc początkowo



składowa tc_SPReg jest równa tc_SPUpper. Warto przy tym pamiętać, że tc_SPUpper wskazuje na górną granicę stosu, tak więc pierwsze słowo na stosie ma adres tc_SPUpper - 2.

Z innych składowych tc_State zawiera status tasku (RUNNING, READY, WAITING), tc_SigWait to informacja, na jaki typ sygnałów dany task oczekuje (tj. są to sygnały wyszczególniane w funkcji Wait()), wreszcie tc_Launch wskazuje na początek programu, który task ma wykonać.

Chcąc sprawdzić ilość i typ tasków w stanie READY oraz WAITING musimy jeszcze wiedzieć, gdzie znajdują się wskaźniki na pierwsze elementy tych list. W strukturze ExecBase składowa TaskReady wskazuje początek listy tasków w stanie READY, składowa TaskWait to wskaźnik na początek listy tasków w stanie WAITING. Jednakże wskaźniki te nie wskazują bezpośrednio na strukturę Node, lecz na strukturę typu List! Dopiero składowa lh_Head tej struktury wskazuje na pierwszy task listy. Wystarczy teraz tylko pamiętać, że przeszukanie list tasków należy przeprowadzić wyłączając wcześniej przerwania, a wskaźniki do odnalezionych tasków zapisywać w tablicy. Dołączony program postępuje według tego algorytmu, jednak należą się dwa słowa komentarza. Po pierwsze, tablica na wskaźniki do tasków jest przewidziana na 100 pozycji. Jeśli jakimś cudem byłoby więcej tasków w systemie (???), wtedy program przekroczy obszar tablicy i zacznie zapisywać dane poza nią. 100 tasków to jednak taka ilość, że jest to sytuacja dość abstrakcyjna. Może jednak pojawić się inny błąd. Otóż przeszukanie list tasków następuje przy wyłączonych przerwaniach, jednak wydruk danych odbywa się po uruchomieniu przerwań, może więc zdarzyć się sytuacja, w której task o adresie przechowywanym w naszej tablicy zostanie usunięty z pamięci. W takim przypadku nie ma żadnej gwarancji, że dane wskazywane przez zapamiętany wskaźnik będą poprawne. Ten typ błędu nie jest na szczęście groźny w skutkach, może jedynie spowodować wyprowadzenie niewłaściwych informacji na ekran.

Dołączany program był kompilowany przez kompilator Aztec C 5.2a z parametrami wywołania:

```
cc showtasks
ln showtasks -lc
```

Uruchomienie programu na kompilatorze SAS C wymaga jedynie kosmetycznych zmian. ❑

```
#include <exec/execbase.h>


/* Deklaracja funkcji. */
void PrintTaskData( register struct Task *task );
/* Wskaźnik na strukturę ExecBase. */
struct ExecBase *SysBase;



main()
{
  UBYTE loop, firstwaiting, lasttask;
  struct Task *current;
  struct Task *tasks[100];

  loop = 0;
  lasttask = 0;
  firstwaiting = 0;
  /* Wyłączenie przerwań. */

  Disable();

  /* Task w stanie RUNNING. */

  tasks[ loop++ ] = SysBase->ThisTask;

  /* Przeszukanie listy tasków w stanie READY. */

  current = (struct Task *) SysBase->TaskReady.lh_Head;

  while ( current->tc_Node.ln_Succ )
   {
      tasks[ loop++ ] = current;
      current = (struct Task *) current->tc_Node.ln_Succ;
   }

  /* Gdzie w tablicy rozpoczyna się lista tasków w stanie WAITING. */

  firstwaiting = loop;

  /* Przeszukanie listy tasków w stanie WAITING. */

  current = (struct Task *) SysBase->TaskWait.lh_Head;

  while ( current->tc_Node.ln_Succ )
   {
      tasks[ loop++ ] = current;
      current = (struct Task *) current->tc_Node.ln_Succ;
   }

  /* Gdzie jest pierwszy wolny element tablicy. */

  lasttask = loop;

  /* Włączenie przerwań. */

  Enable();

  /* Wydruk informacji o taskach. */

  for ( loop=0; loop < lasttask; loop++ )
   {
     if ( loop == 0 )
       printf( "----------------------------------Task in RUNNING state:" );
     else
      if ( loop == 1 && firstwaiting != 1 )
        printf( "----------------------------------Task(s) in READY state:" );
      else
       if ( loop == firstwaiting )
         printf( "----------------------------------Task(s) in WAITING state:" );

     PrintTaskData( tasks[loop] );
   }

  printf( "" );
}


/* Funkcja drukuje podstawowe informacje o tasku wskazywanym
przez parametr task. */

void PrintTaskData( register struct Task *task )
{
  printf( "%24s:  PRI: %3d,  STACK: %8x [%4x],  SIG:
                                      %8x", task->tc_Node.ln_Name,
          task->tc_Node.ln_Pri, task->tc_SPLower,
          (ULONG) task->tc_SPUpper - (ULONG) task->tc_SPLower
                                      - 2, task->tc_SigWait );
}
```

UWAGA! Linie wytłuszczone należy wpisać w jednym ciągu.



ciąg dalszy ze strony 00

stała się prawdziwym hitem. W rozmowie z p. Stumpfem, którą odbyliśmy następnego dnia, ujawnił on, że kampania telewizyjna CD32 pochłania 80% budżetu niemieckiego oddziału firmy. Nie jest to specjalnie pocieszająca informacja dla polskiego użytkownika, gdyż CD32 na naszym rynku praktycznie nie istnieje i - z powodu wysokich cen oprogramowania (czyt. gier) - specjalnie dużych szans na nim nie ma. Prócz tego, taka polityka firmy utrwala opinię Amigi jako maszynki do gier i spycha na margines rynku inne modele (A4000). Zapytany o działalność firmy na rynkach Europy Wschodniej, p. Stumpf stwierdził, że rekordy sprzedaży bije rynek polski. Rocznie sprzedaje się ok. 25000 sztuk komputerów, głównie A500, A600 i ostatnio A1200. Dane te są zaniżone, gdyż obejmują tylko sprzedaż prowadzoną przez oficjalnych dealerów, a całkowicie pomijają import prywatny i import małych firm. Jako ciekawostkę p. Stumpf podał fakt, że firma ma poważne kłopoty z zaspokojeniem popytu polskich dealerów na Commmodore 64. Wywołało to chóralny śmiech na sali konferencyjnej. Wśród uczestników konferencji zauważyliśmy m.in. p. Freiherr von Gravenreuth, znanego w RFN prawnika zajmującego się problematyką ochrony praw autorskich oprogramowania, sławnego jako najskuteczniejszy pogromca niemieckich piratów. Rozważaliśmy nawet możliwość zaproszenia go do Warszawy na giełdę przy ul. Grzybowskiej, lecz pomysł ten szybko upadł. Nie chcieliśmy obciążać swoich sumień ryzykiem spowodowania zawału serca u prawnika.

Około godziny jedenastej konferencja dobiegła końca i wypadało zobaczyć, co dzieje się w hali głównej. Pierwszym wrażeniem było uczucie potwornego tłoku. 11000 metrów kwadratowych brzmi może imponująco, ale przy takiej ilości wystawców i naporze zwiedzających oraz kupujących (szczególnie w drugim dniu imprezy) nie sposób było się normalnie poruszać. Pomimo dużej ilości wystawiających, produktów tak naprawdę nowych nie było zbyt dużo. Większość firm prezentowała wyroby znajdujące się już od pewnego czasu na rynku lub tylko lekko ulepszone ich wersje.

Znany niemiecki dystrybutor sprzętu i oprogramowania, firma Amiga Oberland zainaugurowała sprzedaż nowego, rewelacyjnego edytora tekstów Final Writer będącego kolejnym udoskonaleniem znajdującego się od pewnego czasu na rynku edytora FINAL COPY II. Nowa wersja edytora została wzbogacona o elementy DTP. Kolejną nowością tej firmy był Digital Audio Machine (DAM), przeznaczony do nagrywania i odtwarzania dźwięku na twardym dysku (w jakości CD). DAM jest wyposażona w dwa analogowe WE/WY umożliwiające sampling z częstotliwością próbkowania do 51 kHz w rozdzielczości 16 bitów. Opcjonalnie można na niej zamontować dwa cyfrowe WE/WY (takie same jak w karcie firmy Macrosystem - Maestro). DAM jest oczywiście wyposażona w odpowiednie oprogramowanie.

Monachijska firma BSC pokazała kilka swoich produktów, z których większość od pewnego czasu znajduje się już w sprzedaży. Nowością była 24-bitowa karta graficzna Graffity wykorzystująca możliwości slotu ZORRO III. Prócz tego, można było obejrzeć kartę multi I/O i kontroler pozwalający na uruchomienie z Amigą napędu CD-ROM standadu AT-Bus. Ciekawostką była karta ISDN-Master pozwalająca na pracę w niemieckiej telefonicznej sieci cyfrowej. Przy jej pomocy można osiągnąć prędkości transmisji rzędu 57000 bps. BSC jest na niemieckim rynku znana z produktów, które charakteryzują się wysoką jakością i jeszcze wyższą ceną. Wyżej wymienione urządzenia nie stanowią pod tym względem wyjątku.

Na znajdującym się na honorowym miejscu pośrodku hali stoisku Commodore znajdowało się kilka Amig CD32, 4000 i 1200, do których podłączone były dwie nowości sprzętowe firmy: Karta Ethernet PCMCIA i moduł MPEG. Firma lansowała A1200 jako komputer do obróbki video w zakresie amatorskim, a A4000 jako sprzęt multimedialny. Dwa zestawy sprzętowo-programowe o nazwach Clip Maker (zestaw do robienia czołówek video) i Desktop Dynamite (zestaw prostego składu komputerowego) przedstawiały naszą przyjaciółkę jako komputer do wszystkiego. Bliższe szczegóły na temat w/w sytemów już niebawem.

Niemiecka firma DTM (filia GVP) zaprezentowała na swoim stoisku m.in. system graficzny EGS wraz z programem EGSPaint w oparciu o dwie karty EGS 110/24 i jej tańszą wersję EGS 28-24 Spectrum. Innym nowym produktem do prac wideo jest TBC-Plus - time base corrector dla systemów NTSC,



Zaludnione stanowisko GVP.

PAL, SECAM, zintegrowany z framegrabberem, dekoderem i koderem kodu czasowego SMPTE. Dla posiadaczy A600/1200 firma zaproponowała 16-bitowy sampler montowany w slocie PCMCIA, a posiadaczom A1200 kartę turbo na procesorze MC68030 z zegarem 50Mhz i opcjonalnym kontrolerem SCSI. Również i dla A4000 przygotowano kartę turbo, sercem jej jest procesor MC68040 z zegarem 40MHz i kontrolerem SCSI II. Jej dystrybucję zapowiedziano na koniec tego roku. Dodać można, że GVP wypuściło również G-LOCK-VGA+ dla komputerów klasy IBM PC (cena karty wznosi jednak 2000 DM). O karcie i całym systemie EGS napiszemy w następnych numerach Amigowca.

Ciekawostką była oferta firmy Compedo, która wprawdzie nie sprzedaje żadnych produktów współpracujących bezpośrednio z Amigą, ale na pewno zainteresuje posiadaczy drukarek. Na stoisku tej firmy było dostępnych mnóstwo wszelkiego rodzaju taśm barwiących przeznaczonych do drukowania nie tylko na papierze, ale i na tkaninach oraz materiałach ceramicznych. Specjalne zbiorniczki wielokrotnego użytku sprzedawane przez Compedo pozwalają na znaczną redukcję kosztów druku drukarkami atramentowymi.

Na stanowisku Elctronic Design można było podziwiać w akcji nowy genlock tej firmy. Neptun-Genlock, oprócz typowych funkcji, wyposażony jest dodatkowo w Alpha Channel. ED zainaugurowała także sprzedaż nowego oprogramowania do Frame Machine zawierającego 24-bitowy program malarski.



Najnowszą wersję znanego programu DTP ProPage 5.0 można było obejrzeć na stanowisku niemieckiego dystrybutora Gold Disk - wydawnictwa IPV-Verlag.

Najwięcej nowości zaprezentowała niemiecka firma MacroSystem (twórca pierwszego Flicker-Fixera do A2000). MacroSystem zaprezentował funkcjonujące prototypy nowej wersji 24-bitowej karty graficznej Retina BLT Z3 i zintegrowanej karty video Vlab Motion. Nowa wersja Retiny wykorzystuje w pełni możliwości slotu ZORRO III i jest kompatybilna programowo z wersją poprzednią. Retina, mimo raczej niezbyt nowoczesnych rozwiązań konstrukcyjnych, jest w tej chwili z pewnością najlepiej działającą (dopracowane oprogramowanie i wysoka kompatybilność) kartą graficzną w obszarze cenowym do 1000 DM. Mamy nadzieję, że nowa wersja także będzie zasługiwała na wysokie oceny.

Prawdziwym hitem ma szansę stać się Vlab Motion. Jest on kombinacją digitizera czasu rzeczywistego, genlocka i 24-bitowej karty graficznej. Digitizer jest zdolny do nagrywania obrazu z taśmy video na twardym dysku z prędkością 25 pełnych klatek na sekundę. Stosuje przy tym hardware'ową kompresję w formacie JPEG, która pozwala na zapamiętanie sygnału video w pełnej rozdzielczości PAL. Zintegrowana karta graficzna, pozwala na pracę z obrazem 24-bitowym o rozdzielczości 768*576 punktów (overscan video). Karta posiada wyjścia CVBS i Y/C (SVHS). Całkiem przyzwoite parametry posiada także umieszczony na niej genlock. Jedną z jego ważniejszych zalet jest możliwość zdefiniowania przezroczystości dla pewnego zakresu barw, bez ograniczenia się do jednego koloru. Vlab Motion będzie dostępny w sprzedaży w połowie grudnia b.r.

Trzecią z kolei nowością sprzętową MacroSystem jest karta dźwiękowa Toccata. Jest ona taką samą rewelacją w dziedzinie obróbki dźwięku, jak Vlab Motion w zakresie video. Nawiasem mówiąc, obie karty doskonale ze sobą współpracują i uzupełniają się wzajemnie tworząc kompletny system audio-video. Toccata zawiera 16-bitowy sampler pozwalający na digitalizację dźwięku z maksymalną częstotliwością próbkowania 48 kHz, z jednoczesnym zapisem w czasie rzeczywistym na twardy dysk. Parametry zapisywanego dźwięku także są niezłe: Dynamika 95 dB w zakresie częstotliwości 10 Hz-20 kHz.



Rozmowa z Fredem Fishem.

Karta posiada trzy stereofoniczne wejścia analogowe, jedno wejście mikrofonowe (też stereo) i jedno stereofoniczne wyjście. Sygnały z wszystkich wejść można dowolnie mieszać.

Oferta sprzętowa firmy została wzbogacona ciekawym oprogramowaniem produkcji własnej i firm współpracujących. Do karty Toccata dołączany jest program Samplitude firmy SEK'D. Osobno można nabyć doskonały program graficzny Brillance i zestaw do nagrywania (odtwarzania) video na (z) twardym dysku: DCTV+BAY.

Nową bazą danych MaxonTwist i raytracerem MaxonCinema błysnęła firma Maxon, znana już wcześniej z takich produktów jak: MaxonWord, MaxonCAD i Maxon C++.

Jeśli komuś z Was nie podoba się obudowa Waszego komputera, bądź też uważacie, że zajmuje ona na stole zbyt wiele miejsca, możecie zaopatrzyć się w niemieckiej firmie Micronik w obudowę typu Tower (mini, midi, big). Oferowała ona na swoim stoisku obudowy z własnym zasilaczem do wszystkich typów Amig, często z dodatkowymi slotami. Za pomocą takiej obudowy można bez problemu przerobić pięćsetkę, czy A1200 na semiprofesjonalny komputer pozwalający na korzystanie z kart ZORRO II. Micronik ma też coś ciekawego dla zniecierpliwionych potencjalnych nabywców A4000 TOWER. Pudło do Amigi 4000, oprócz imponującego wyglądu, posiada miejsce na 6 napędów 5,25", 5 napędów 3,5" i rozszerzenie płyty głównej o 7 (!) slotów ZORRO III.

Toccata firmy MacroSystem nie była jedyną 16-bitową kartą muzyczną obecną na imprezie. Firma Omega Datentechnik pokazała kartę o nazwie Aida posiadającą oprócz obligatoryjnych 16-bitowych przetworników C/A i A/C, mikser cyfrowy i analogowy. Konkurencji doczekał się także Scan-Doubler. Monika - bo tak nazywa się produkt Omegi, spełnia taką samą jak SD funkcję.

Gigantyczną porcję oprogramowania zaserwowała firma Stefan Ossowski Schatzruhe specjalizująca się w produkcji taniego (dla zachodniego użytkownika) software. Najciekawszymi ofertami tej firmy były z pewnością: Program kalkulacyjny Turbocalc i Disk Expander - kompresor twardego dysku.

Na stoisku tej firmy honorowym gościem był Fred Fish, sprzedający dyski Public Domain CD opatrzone własnym autografem. Przeprowadziliśmy z nim krótką rozmowę. Kolekcja, którą Fish organizuje i rozpowszechnia od 1985 roku liczy obecnie 930 dysków. Nad powstawaniem kolekcji pracuje on sam, a w dystrybucji pomagają mu tylko trzy osoby. Obecnie Fish pracuje na Amidze 4000/40, przyznał jednak, że w USA jest obecnie brak wszelkiego osprzętu do tego modelu komputera. Zadowalające jest to, że jego cała kolekcja działa poprawnie na Amidze CD32. Zanotował jednak jedną wpadkę, na dysku 919 (w pliku BBBS) znajduje się wirus. Nawiązany przez Amigowca kontakt powinien zaowocować i o nowościach Fisha napiszemy już niebawem.

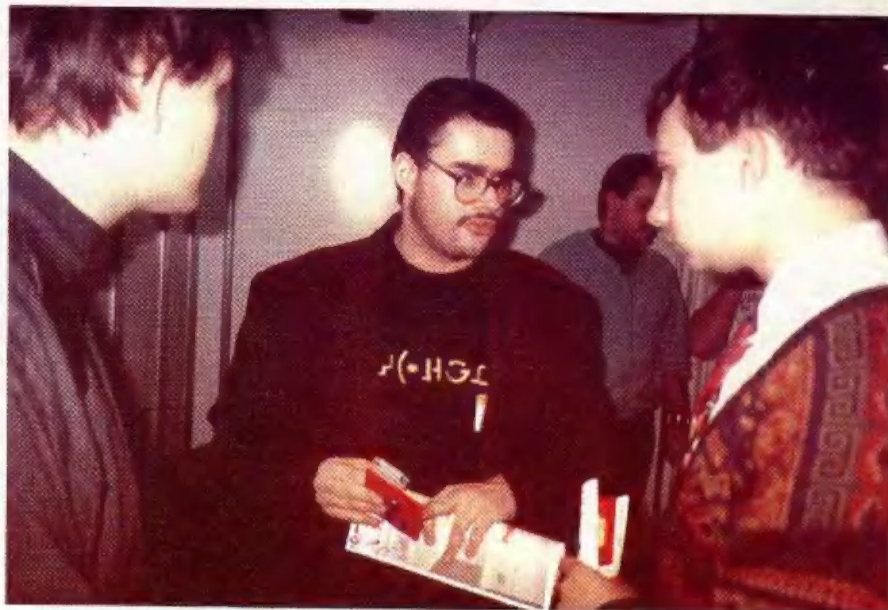
ciąg dalszy na stronie 81



ciąg dalszy ze strony 77

Istniejącą na amigowskim rynku lukę w dziedzinie osprzętu sieciowego wypełni, miejmy nadzieję, znana z produkcji karty graficznej Picasso, firma VillageTronic. Ariadne, tak będzie nazywać się karta sprzedawana razem z licencjonowanym przez Commodore pakietem sieciowym Envoy. Przewidywana cena będzie wyraźnie niższa od podobnych produktów innych firm.

Reklama jest dźwignią handlu. To wyświechtane powiedzenie nie starciło absolutnie na aktualności. Można się było o tym przekonać na stoisku firmy X-Pert, której, moim zdaniem, przereklamowana karta graficzna Merlin sprzedawała się jak radzieckie telewizory w czasach ustroju sprawiedliwości społecznej. Wielu nabywców spotka pewnie rozczarowanie, gdy uruchomią ją w swoim komputerze. Oprogramowanie Merlina jest niedopracowane (szczególnie emulator Workbencha; na A4000 jest on straszliwie powolny), a podane w reklamie zalety nie są wprawdzie nieprawdą, lecz raczej kombinacją sprytnych niedopowiedzeń na zasadzie: Tak, ale ... . Najpoważniejszą wadą Merlina jest niska rozdzielczość w trybie 24-bitowym. Chociaż liczby podane w ukazującej się od roku (sic!) reklamie w Amiga Magazin sugerują możliwość uzyskania rozdzielczości: "Tysiąc z kawałkiem razy jeszcze więcej" (nie podaję dokładnej liczby, bo jeszcze ktoś zapamięta i uwierzy), to są to maksymalne rozmiary ekranu, które można dzięki karcie otworzyć. Widocznych na ekranie jest tylko 644*580 punktów, co dyskwalifikuje Merlina do prac video, chyba, że posiadacie komputer telewizyjny dopasowujący obraz do właściwych romiarów (kosztuje ok. 3 miliardów zł). Jest to tym bardziej dziwne, że X-Pert reklamuje Merlina do prac video i oferuje (ale tylko na papierze) dodatkowo genlock i ditgitizer do karty.



Odbieramy nową wersję X-Copy.

Zupełną rewelacją w zastosowaniach video okazał się produkt firmy Digital Micronics (USA) - Digital BroadCaster 32. Jest to urządzenie służące do montażu i wideo edycji w oparciu o kompresję w standarcie JPEG. Karta oferuje pracę w profesjonalnej jakości obrazu o rozdzielczości 768x586, wyświetlanej z częstotliwością 25 klatek na sekundę. Zakładając, że posiadamy 2GB dysk twardy to długość materiału dla standartu Betacam (obecny system telewizyjny) może wynosić 6-20 minut, a dla prac VHS ok. 40 minut. Digital BroadCaster 32 umożliwia także zastąpienie zwykłego poklatkowego zgrywania obrazu przez pracę z animacją "na żywo" z komputera. Podobną funkcję pełni Personal Animation Recorder firmy DPS, który, jak się dowiedzieliśmy od niemieckiego dystrybutora, w grudniu ma być dostępny również dla systemu PAL.



Russell Neale prezentuje nowy produkt GVP, kartę TBC.

Mimo iż z relacji wynika, że pojawiło się sporo nowych rzeczy, w rzeczywistości jednak większość ofert istniała jedynie na papierze lub w fazie prezentacji prototypu. Również organizacja targów pozostawiała wiele do życzenia. Plan ekspozycji posiadał bardzo nieczytelne oznaczenia stanowisk. Hala wystawowa zdawała się być niedostosowana do wielkiej ilości zwiedzających. Jednym słowem raj dla kieszonkowców. Dopuszczenie do targów firm oferujących sprzęt klonowy również nie wpływa pozytywnie na obraz Amigi. Naszym zdaniem Commodore powinien zastanowić się nad wyborem innego organizatora swoich imprez w przyszłości oraz nad zmianą polityki reklamowej, szczególnie w stosunku do najwyższych modeli Amig. ❑





TEN ZESTAW MOŻE BYĆ TWÓJ ZA JEDYNE 65 tys. zł





Czerwony kupon przesłany wraz z zamówieniem na zestaw na adres redakcji uprawnia do wzięcia udziału w losowaniu nagród (5 gier komputerowych)

+ AMIGOWIEC NIESPODZIANKA



Do wygrania 5 gier

KUPON KONKURSOWY

imię ....

nazwisko ....

adres ....



Swoje zamówienie wyślij na adres:

ALFIN sp. z o.o.

ul. Świętojańska 2/7
85-017 Bydgoszcz
tel. 28-79-20
fax 22-64-03